DEEP LEARNING FOR TOXICITY AND DISEASE PREDICTION

EDITED BY: Ping Gong, Chaoyang Zhang and Minjun Chen

PUBLISHED IN: Frontiers in Genetics, Frontiers in Physiology,
Frontiers in Bioengineering and Biotechnology
and Frontiers in Plant Science

frontiers Research Topics

ISSN 1664-8714
ISBN 978-2-88963-632-7
DOI 10.3389/978-2-88963-632-7

About Frontiers

Frontiers is more than just an open-access publisher of scholarly articles: it is a pioneering approach to the world of academia, radically improving the way scholarly research is managed. The grand vision of Frontiers is a world where all people have an equal opportunity to seek, share and generate knowledge. Frontiers provides immediate and permanent online open access to all its publications, but this alone is not enough to realize our grand goals.

Frontiers Journal Series

The Frontiers Journal Series is a multi-tier and interdisciplinary set of open-access, online journals, promising a paradigm shift from the current review, selection and dissemination processes in academic publishing. All Frontiers journals are driven by researchers for researchers; therefore, they constitute a service to the scholarly community. At the same time, the Frontiers Journal Series operates on a revolutionary invention, the tiered publishing system, initially addressing specific communities of scholars, and gradually climbing up to broader public understanding, thus serving the interests of the lay society, too.

Dedication to Quality

Each Frontiers article is a landmark of the highest quality, thanks to genuinely collaborative interactions between authors and review editors, who include some of the world's best academicians. Research must be certified by peers before entering a stream of knowledge that may eventually reach the public - and shape society; therefore, Frontiers only applies the most rigorous and unbiased reviews.
Frontiers revolutionizes research publishing by freely delivering the most outstanding research, evaluated with no bias from both the academic and social point of view. By applying the most advanced information technologies, Frontiers is catapulting scholarly publishing into a new generation.

What are Frontiers Research Topics?

Frontiers Research Topics are very popular trademarks of the Frontiers Journals Series: they are collections of at least ten articles, all centered on a particular subject. With their unique mix of varied contributions from Original Research to Review Articles, Frontiers Research Topics unify the most influential researchers, the latest key findings and historical advances in a hot research area! Find out more on how to host your own Frontiers Research Topic or contribute to one as an author by contacting the Frontiers Editorial Office: researchtopics@frontiersin.org

DEEP LEARNING FOR TOXICITY AND DISEASE PREDICTION

Topic Editors:
Ping Gong, U.S. Army Engineer Research and Development Center, United States
Chaoyang Zhang, University of Southern Mississippi, United States
Minjun Chen, National Center for Toxicological Research, U.S. Food and Drug Administration, United States

Citation: Gong, P., Zhang, C., Chen, M., eds. (2020). Deep Learning for Toxicity and Disease Prediction. Lausanne: Frontiers Media SA. doi: 10.3389/978-2-88963-632-7

Table of Contents

frontiers
in Genetics

EDITORIAL
published: 26 February 2020
doi: 10.3389/fgene.2020.00175

Editorial: Deep Learning for Toxicity and Disease Prediction

Ping Gong[1]*, *Chaoyang Zhang*[2] *and Minjun Chen*[3]

[1] *Environmental Laboratory, U.S. Army Engineer Research and Development Center, Vicksburg, MS, United States,* [2] *School of Computing Sciences and Computer Engineering, University of Southern Mississippi, Hattiesburg, MS, United States,* [3] *Division of Bioinformatics and Biostatistics, National Center for Toxicological Research, U.S. Food and Drug Administration, Jefferson, AR, United States*

Keywords: deep learning, disease diagnosis or prognosis, chemical toxicity prediction, deep neural networks, conventional machine learning

OPEN ACCESS

Edited and reviewed by:
Douglas Mark Ruden,
Wayne State University, United States

****Correspondence:***
Ping Gong
ping.gong@usace.army.mil

Specialty section:
This article was submitted to Toxicogenomics, a section of the journal Frontiers in Genetics

Received: *22 January 2020*
Accepted: *13 February 2020*
Published: *26 February 2020*

Citation:
Gong P, Zhang C and Chen M (2020) Editorial: Deep Learning for Toxicity and Disease Prediction. Front. Genet. 11:175. doi: 10.3389/fgene.2020.00175

Editorial on the Research Topic

Deep Learning for Toxicity and Disease Prediction

Deep learning (DL), alsocalled deep structured learning or hierarchical learning, is an important subset of machine learning (ML). The distinction between DL and conventional "shallow" ML is that DL algorithms allow computational models composed of multiple processing layers to be fed with raw data and automatically learn multiple levels of abstract representations of data for detection and classification (LeCun et al., 2015). The history of DL can be traced back to the 1940s when the first neural network model was developed (McCulloch and Pitts, 1943). It wasn't until recently that DL evolved into and reemerged as a prominent discipline within the artificial intelligence domain, thanks to such revolutionary advances as backpropagation, parallel computing with GPUs, availability of massive labeled data, improved architectures, robust optimizers, regularization techniques, and activation functions (see https://www.import.io/post/history-of-deep-learning/ and https://beamandrew.github.io/deeplearning/2017/02/23/deep_learning_101_part1.html for more info). Over the past decade DL has regained popularity and has been successfully applied to such diverse fields as image (Zeiler and Fergus, 2014) and speech (Hinton et al., 2012) recognition, visual art (Huang et al., 2016) and natural language (Xiong et al., 2016) processing, drug discovery (Gawehn et al., 2016), chemical toxicity prediction (Mayr et al., 2016), and computational biology (Angermueller et al., 2016). For instance, deep convolutional neural networks (CNNs) have brought about breakthroughs in computer vision and pattern recognition (Krizhevsky et al., 2012), whereas recurrent neural networks have shed light on sequential data such as text mining and speech applications (Hinton et al., 2012).

Despite great success, there remain many technical challenges, one of which is how to integrate or transform subject-specific knowledge in order to adapt to DL algorithms and improve outcomes. Technical hurdles exist in data preprocessing, model selection (e.g., feedforward, convolutional, or recurrent networks), parametric function approximation (e.g., initialization strategies, activation functions, architecture, and learning techniques), and model regularization and optimization. This Research Topic addresses these challenges and hurdles with a specific focus on the application of DL algorithms to chemical toxicity prediction and disease diagnosis, which has not been adequately explored (Mayr et al., 2018; Xu et al., 2019). As a result, 11 manuscripts were accepted in four participating journals: 7 in Frontiers in Genetics (Zhang L. et al.; Hu et al.; Jia et al.; Luo et al.; Xie et al.; Zhang X. et al.; Ji et al.), 2 in Frontiers in Plant Science (Fuentes et al.; Lin et al.), 1 in Frontiers in Physiology (Idakwo et al.), and 1 in Frontiers in Bioengineering and Biotechnology (Matsuzaka and Uesawa). These papers are well-split between human (Zhang L. et al.; Jia et al.; Luo et al.; Xie et al.; Zhang X. et al.) or plant (Fuentes et al.; Lin et al.) disease diagnosis and chemical toxicity (Matsuzaka and Uesawa; Idakwo et al.) or drug efficacy (Hu et al.; Ji et al.) prediction. CNN architecture dominated these studies, except three where autoencoder (Zhang L. et al.; Hu et al.) or XGBoost (Ji et al.) was employed. The input data varied from images (Fuentes et al.; Lin et al.; Xie et al.) or converted images (Matsuzaka and Uesawa) to gene mutations (Luo et al.), chemical molecular descriptors (Hu et al.; Idakwo et al.), phenotypes (Jia et al.), physical examination records (Zhang X. et al.), and mixtures of different data profiles such as multi-omics data (Zhang L. et al.), chemical structures, human phenotypes, pathways, protein targets, and protein–protein interactions (Ji et al.).

As summarized below, this collection of original research papers presents a significant amount of progress made in the above-mentioned scope of the Research Topic:

Development of novel DL-based tools: Autoencoder-based classification models were developed to identify ultra-high risk prognostic subgroups of neuroblastoma (Zhang et al.) or distinguish drug-like compounds from common compounds (Hu et al.). Luo et al. demonstrated that a CNN-based deepDriver could learn information within somatic mutation data and similarity networks simultaneously to enhance the prediction of cancer driver genes. A CNN-based, pixel-level semantic segmentation model was built for quantitative assessment of the severity of powdery mildew in cucumber leaves, achieving an average pixel accuracy of 96% (Lin et al.). Xie et al. applied both CNN- and autoencoder-based DL and transfer learning techniques to automatically extract high-level abstract features from breast cancer histopathological images, which led to a significant improvement in cancer diagnosis. Zhang X. et al. reported a novel GroupNet model for multi-label chronic disease classification that outperformed other DL (e.g., AlexNet) and conventional ML (e.g., SVM) models.

Optimization of existing DL-based tools: Fuentes et al. presented a two-tiered diagnosis system to address high false positive rates caused by class unbalance and variation. The system consists of a primary diagnosis unit that detects a set of bounding boxes that likely contain a disease in the image, a secondary diagnosis unit that verifies bounding boxes detected from the primary diagnosis unit using independent CNN classifiers trained with respect to each class, and an integration unit that combines the results from the primary and secondary units to effectively recognize 10 different types of diseases and pests in tomato. This system showed an improved recognition rate of 96%, 13% higher than previous work (Fuentes et al., 2017). Matsuzaka and Uesawa refined DeepSnap, a DL-based tool for quantitative structure-activity relationship (QSAR) analysis previously developed by Uesawa (2018), through optimizing such parameters as the number of molecules per Structure Data File (SDF), zoom factor percentage, atom size for van der Waals percentage, bond radius, minimum bond distance, and bond tolerance. The DeepSnap with an optimal set of parameter values generated the best performing models.

Choosing between DL and conventional ML (cML): Despite revolutionary breakthroughs, DL does not always provide better performance or superior solutions to any specific problem than cML. Such cML as Logistic Regression (LR), Random Forest (RF), and Naive Bayes (NB) were employed along with Deep Neural Network (DNN) to train classifiers with excellent precision (≥98%) and recall (up to 95%) for rare disease diagnosis implemented in a Rare Disease Auxiliary Diagnosis system (Jia et al.). Idakwo et al. presented a case study where DNN and RF were compared with and without parametric optimization in terms of QSAR-based chemical toxicity prediction. Ji et al. compared XGBoost, a cML algorithm, with DeepSynergy, a DL algorithm, and other cML algorithms (e.g., RF, LR, and NB), and concluded that XGBoost outperformed other classifiers in both stratified five-fold cross-validation and independent validation in identifying synergistic or antagonistic drug combinations. These studies suggest that in the absence of large amounts of training samples (e.g., in the 100 or 1,000 k range), cML may be an alternative superior to DL in performance, as cML is less likely to over-fit and often computationally less costly. Even with available big data, DL algorithms need to be optimized to achieve outstanding performance (Fuentes et al.; Idakwo et al.; Matsuzaka and Uesawa). Furthermore, transfer learning was used in conjunction with DL to train a neural network model on a problem similar to the one being solved (Xie et al.; Matsuzaka and Uesawa).

Data preprocessing: In order to take advantage of the power of CNN, Matsuzaka and Uesawa converted SMILES text files into SDF image files, whereas Zhang X. et al. transformed physical examination records into multi-label class data using binary relevance and label powerset methods. Data rebalance techniques (Hu et al.; Xie et al.) and focal loss (Zhang X. et al.) or stratification (Ji et al.; Idakwo et al.) strategies were often performed to overcome the influence of skewed class distribution. Data preprocessing played a critical role in improving performance of DL- or cML-based classification.

This collection of contributions highlights not only the promising outlook of DL applications in disease diagnosis and toxicity prediction, but also the necessity of optimizing DL algorithms in order to achieve superior outcomes. Given the remarkable success of DL application in classification problems, the focus of future efforts may now shift to quantification problems.

AUTHOR CONTRIBUTIONS

PG proposed and edited this Research Topic. CZ and MC co-edited this Research Topic. All authors made a substantial, direct and intellectual contribution to this Editorial, and approved it for publication.

FUNDING

This work was supported in part by funding provided to PG from the U.S. Army Environmental Quality and Installation Program.

ACKNOWLEDGMENTS

Permission was granted by the Chief of Engineer, U.S. Army Corps of Engineers to publish this paper. We thank Drs. Ioannis P. Androulakis, Nora L. Nock, and Alfredo Pulvirenti for editing three papers in this collection.

REFERENCES

Angermueller, C., Pärnamaa, T., Parts, L., and Stegle, O. (2016). Deep learning for computational biology. *Mol. Syst. Biol.* 12:878. doi: 10.15252/msb.20156651

Fuentes, A., Yoon, S., Kim, S. C., and Park, D. S. (2017). A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. *Sensors* 17:E2022. doi: 10.3390/s17092022

Gawehn, E., Hiss, J. A., and Schneider, G. (2016). Deep learning in drug discovery. *Mol. Inform.* 35:3–14. doi.org/10.1002/minf.201501008

Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A.-R., Jaitly, N., et al. (2012). Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups. *Signal. Process. Mag. IEEE* 29, 82–97. doi: 10.1109/MSP.2012.2205597

Huang, S., Li, X., Zhang, Z., He, Z., Wu, F., Liu, W., et al. (2016). Deep learning driven visual path prediction from a single image. *IEEE Trans. Image Process.* 25, 5892–5904. doi: 10.1109/TIP.2016.2613686

Krizhevsky, A., Sutskever, I., and Hinton, G. (2012). ImageNet classification with deep convolutional neural networks. *Proc. Adv. Neural Inform. Process. Syst.* 25, 1090–1098. doi: 10.1145/3065386

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature* 521, 436–444. doi: 10.1038/nature14539

Mayr, A., Klambauer, G., Unterthiner, T., and Hochreiter, S. (2016). DeepTox: toxicity prediction using deep learning. *Front. Environ. Sci.* 3:80. doi: 10.3389/fenvs.2015.00080

Mayr, A., Klambauer, G., Unterthiner, T., Steijaert, M., Wegner, J. K., Ceulemans, H., et al. (2018). Large-scale comparison of machine learning methods for drug target prediction on ChEMBL. *Chem. Sci.* 9, 5441–5451. doi: 10.1039/c8sc00148k.

McCulloch, W. S., and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *Bull. Math. Biophys.* 5, 115–133.

Uesawa, Y. (2018). Quantitative structure–activity relationship analysis using deep learning based on a novel molecular image input technique. *Bioorg. Med. Chem. Lett.* 28, 3400–3403. doi: 10.1016/j.bmcl.2018.08.032

Xiong, C., Merity, S., and Socher, R. (2016). Dynamic memory networks for visual and textual question answering. *arXiv[Preprint]*. arXiv:1603.01417.

Xu, J., Xue, K., and Zhang, K. (2019). Current status and future trends of clinical diagnoses via image-based deep learning. *Theranostics* 9, 7556–7565. doi: 10.7150/thno.38065

Zeiler, M. D., and Fergus, R. (2014). "Visualizing and understanding convolutional networks," *Computer Vision–ECCV 2014*, eds D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars (Cham: Springer), (Heidelberg; Berlin: Springer), 818–833.

Disclaimer: The content is solely the responsibility of the authors and does not necessarily represent the official views of U.S. Army Corps of Engineers and U.S. Food and Drug Administration.

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

frontiers in Plant Science

ORIGINAL RESEARCH
published: 29 August 2018
doi: 10.3389/fpls.2018.01162

High-Performance Deep Neural Network-Based Tomato Plant Diseases and Pests Diagnosis System With Refinement Filter Bank

Alvaro F. Fuentes[1], *Sook Yoon*[2], *Jaesu Lee*[3] *and Dong Sun Park*[4,5]*

[1] Department of Electronics Engineering, Chonbuk National University, Jeonju, South Korea, [2] Department of Computer Engineering, Mokpo National University, Muan, South Korea, [3] Department of Agricultural Engineering, National Institute of Agricultural Sciences (RDA), Jeonju, South Korea, [4] College of Computer Science and Information Engineering, Tianjin University of Science and Technology, Tianjin, China, [5] Division of Electronics and Information Engineering, Chonbuk National University, Jeonju, South Korea

OPEN ACCESS

Edited by:
Minjun Chen,
National Center for Toxicological Research (FDA), United States

Reviewed by:
Konstantinos Ferentinos,
Hellenic Agricultural Organization–ELGO, Greece
Tieliu Shi,
East China Normal University, China

***Correspondence:**
Dong Sun Park
dspark@jbnu.ac.kr

Specialty section:
This article was submitted to Bioinformatics and Computational Biology,
a section of the journal Frontiers in Plant Science

Received: *16 April 2018*
Accepted: *23 July 2018*
Published: *29 August 2018*

Citation:
Fuentes AF, Yoon S, Lee J and Park DS (2018) High-Performance Deep Neural Network-Based Tomato Plant Diseases and Pests Diagnosis System With Refinement Filter Bank. Front. Plant Sci. 9:1162. doi: 10.3389/fpls.2018.01162

A fundamental problem that confronts deep neural networks is the requirement of a large amount of data for a system to be efficient in complex applications. Promising results of this problem are made possible through the use of techniques such as data augmentation or transfer learning of pre-trained models in large datasets. But the problem still persists when the application provides limited or unbalanced data. In addition, the number of false positives resulting from training a deep model significantly cause a negative impact on the performance of the system. This study aims to address the problem of false positives and class unbalance by implementing a Refinement Filter Bank framework for Tomato Plant Diseases and Pests Recognition. The system consists of three main units: First, a Primary Diagnosis Unit (Bounding Box Generator) generates the bounding boxes that contain the location of the infected area and class. The promising boxes belonging to each class are then used as input to a Secondary Diagnosis Unit (CNN Filter Bank) for verification. In this second unit, misclassified samples are filtered through the training of independent CNN classifiers for each class. The result of the CNN Filter Bank is a decision of whether a target belongs to the category as it was detected (True) or not (False) otherwise. Finally, an integration unit combines the information from the primary and secondary units while keeping the True Positive samples and eliminating the False Positives that were misclassified in the first unit. By this implementation, the proposed approach is able to obtain a recognition rate of approximately 96%, which represents an improvement of 13% compared to our previous work in the complex task of tomato diseases and pest recognition. Furthermore, our system is able to deal with the false positives generated by the bounding box generator, and class unbalances that appear especially on datasets with limited data.

Keywords: plant diseases, detection, deep neural networks, filter banks, false positives

INTRODUCTION

Plant diseases cause major production and economic loses in the agriculture area. It is nowadays considered as a big issue in the modern agricultural production. Plant protection, in particular, the protection of crops against diseases, has a special role in achieving a higher demand for food and are directly related to the human well-being. Along with the worldwide population, the availability per capita of food is expected to be increased for the next years (Pinstrup-Andersen, 2002). The demand for food is influenced by factors such as the population growth, income levels, urbanization, lifestyles, and preferences (Savary et al., 2012). Therefore, the importance of a proper control during the production process has played an important role in recent times.

An accurate estimation of diseases and pest in plants remains a challenge in the scientific community (Donatelli et al., 2017). Diseases and pest in plants can be generated by several causes (Fuentes et al., 2016) and show different variations throughout their infection status (Fuentes et al., 2017a). Bacteria, fungus, viruses, and insects may result in plant disease and damage (Sankaran et al., 2010). Once infected, a plant develops several symptoms that, if spread, can cause a significant impact on the entire crop. Traditional methods to treat diseases in plants include the use of pesticides. However, an excessive use of pesticides not only increases the cost of production but can also cause an impact on the quality of food. Consequently, a precise estimation of disease incidence, disease severity, and the negative effects of diseases on the quality and quantity of agriculture are important for crop field, horticulture, plant breeding, and improving fungicide efficacy, as well as for plant research (Mahlein, 2016). Monitoring of the growing conditions and detecting diseases in plants is, therefore, critical for sustainable agriculture. In some way, an early detection of suspicious areas in the plant may prevent several economic loses and facilitate the control through appropriate management strategies to increase productivity (Johannes et al., 2017).

Recent interest in neural networks for several areas, and especially their potential applications in agriculture, has fueled the growth of efficient autonomous systems and their application to real problems. Such applications strongly motivate our research in the recognition of pathologies that affect plants, and particularly tomato plants, and at the same time provide a strategy to develop better recognition techniques.

Our previous work (Fuentes et al., 2017b) introduced a detector based on Deep Learning for Tomato Diseases and Pest Recognition, which simultaneously performs the localization and diagnosis of nine different types of diseases and pests. In comparison with other techniques, our system shows the following advantages: (1) It uses images taken in the real field, therefore, we avoid the process of collecting samples and analyzing them in the laboratory; (2) It considers the possibility that a plant can be affected simultaneously by several pathologies in the same sample; (3) It uses images captured by different camera devices with various resolutions; (4) It can efficiently deal with different illumination conditions, size of objects, and background variations, etc.; (5) It provides a practical application in real time that can be used in the field without using expensive and complex technology.

Although the task has been effectively achieved with satisfactory results. We believe that there is still a room that needs to be addressed for this practical application. In fact, we consider that this task remains challenging due to the following conditions: (1) The limited training data with significant unbalanced distribution on the annotated data makes the learning process more biased toward classes with more samples and variations (e.g., leaf mold, canker, plague) while resulting in lower performance in scattered annotated classes with fewer samples (e.g., gray mold, low temperature, powdery mildew). We called this issue a "*class unbalance*" problem. (2) The discrepancy between the classes due to the inter- and intra-class variations results in a high number of false positives that, in fact, limits the system to achieve higher accuracy in this complex recognition task. Consequently, when developing an efficient plant diseases recognition system, it is essential to deal with those problems.

Following our previous approach (Fuentes et al., 2017b), the proposed system uses a refinement diagnosis strategy, which addresses the aforementioned problems, while achieving a higher recognition rate. The main contributions of this paper are summarized as follows. (1) We propose a diagnosis system for an effective recognition of diseases and pests of tomato plants. A primary diagnosis unit detects a set of bounding boxes that are likely containing a disease in the image, then a secondary diagnosis unit verifies bounding boxes detected from the primary diagnosis unit using independent CNN classifiers trained with respect to each class and, finally, an integration unit combines the results from the primary and secondary units to effectively recognize 10 different types of diseases and pests of tomato plant. (2) We introduce a strategy for dealing with false positives generated by object detection networks, and class unbalances problems that work especially on datasets with limited data. (3) By implementing this approach, we are able to obtain a recognition rate of approximately 96% which represents an improvement of 13% compared to our previous work (Fuentes et al., 2017b) in the complex task of tomato plant diseases and pest recognition. It is important to emphasize that our work contrasts with other disease classification-based works (Kawasaki et al., 2015; Mohanty et al., 2016; Sladojevic et al., 2016; Amara et al., 2017; Ferentinos, 2018; Liu et al., 2018), in that, it is a detection-based approach that provides the class and location instances of a particular disease in the image. Furthermore, it uses images from the Tomato Diseases and Pest Recognition Dataset (Fuentes et al., 2017b), which are collected in different field scenarios with real conditions (lighting, background, size, etc.) using several camera devices.

The remainder of this paper is organized as follows. A detailed review of works related to our approach is presented in section Related Works. Section Diagnosis System with Refinement Filter Bank introduces the technical details of our diagnosis system. In section Experimental Results, the experimental results show the performance of our system in the task of tomato diseases and pests recognition. Finally, in Section Conclusion and Future Works, we conclude the paper and mention our future works.

RELATED WORKS

In this section, we first introduce methods based on neural networks for object detection and recognition. Then, we review some techniques used for detecting anomalies in plants and, finally, investigate advances in false positives reduction.

Image-Based Object Detection and Feature Extractors

Recent years have seen an explosion of visual media available through the internet. This large volume of data has brought new opportunities and challenges for neural network applications. Since the first application of Convolutional Neural Networks (CNN) on the image classification task in the ImageNet Large Scale Visual Recognition Competition 2012 (ILSVRC-2012) (Russakovsky et al., 2015) by AlexNet (Krizhevsky et al., 2012), a CNN composed of 8 layers demonstrated an outstanding performance compared to traditional handcrafted-based computer vision algorithms (Russakovsky et al., 2015). Consequently, in the last few years, several deep neural network architectures have been proposed with the goal of improving the accuracy in the same task.

Object detection and recognition have played an important issue in recent years. In the case of detecting particular categories, earlier applications focused on classification from object-centric images (Russakovsky et al., 2012). Where the goal is to classify an image that likely contains an object in it. However, the new dominant paradigm is not only to classify but also precisely localize objects in the image (Szegedy et al., 2013). Consequently, current state-of-the-art object methods for object detection are mainly based on deep CNNs (Russakovsky et al., 2015). They have been categorized into two types: two-stage and one-stage methods. Two-stage methods are commonly related to the Region-based Convolutional Neural Networks, such as Faster R-CNN (Ren et al., 2016), Region-based Fully Connected Network (R-FCN) (Dai et al., 2016). In these frameworks, a Region Proposal Network (RPN) generated a set of candidate object locations in the first stage, and the second stage classifies each candidate location as one of the classes or background using a CNN. It uses a deep network to generate the features that are posteriorly used by the RPN to extract the proposals. In addition to systems based on region proposals, one-stage frameworks have been also proposed for object detection. Most recently SSD (Liu et al., 2016), YOLO (Redmon et al., 2015) and YOLO V2 (Redmon and Farhadi, 2017) have demonstrated promising results, yielding real-time detectors with accuracy similar to two-stage detectors.

Over the last few years, it has been also demonstrated that deeper neural networks have achieved higher performance compared to simple models in the task of image classification (Russakovsky et al., 2015). However, along with the significant performance improvement, the complexity of deep architectures has been also increased, such as VGG (Simonyan and Zissermann, 2014), ResNet (He et al., 2016), GoogLeNet (Szegedy et al., 2015), ResNeXt (Xie et al., 2017), DenseNet (Huang et al., 2017), Dual Path Net (Chen et al., 2017) and SENet (Hu et al., 2017), etc. As a result, deep artificial neural networks often have far more trainable model parameters than the number of samples they are trained on (Zhang et al., 2017). Despite using large datasets, neural networks are prone to overfitting (Pereyra et al., 2017). On the other hand, several strategies have been applied to improve performance in deep neural networks. For example, data augmentation to increase the number of samples (Bloice et al., 2017), weights regularization to reduce model overfitting (Van-Laarhoven, 2017), randomly dropping activations with Dropout (Srivastava et al., 2014), batch normalization (Ioffe and Szegedy, 2015). Although these strategies have proven to be effective in large networks, the lack of data or class unbalances problems for several applications are still a challenge to deal with. There is no a certain way yet of understanding the complexity of artificial neural networks for their application to any problem. Therefore, the importance of developing strategies that are designed specifically for applications that include limited data and class unbalance issues. In addition, depending on the complexity of the application, the challenge nowadays is to design deep learning methods that can perform a complex task while maintaining a lower computational cost.

Anomaly Detection in Plants

The problem of plant diseases is an important issue that is directly related to the food safety and well-being of the people. Diseases and pest affect food crops, that in turn causes significant losses in the farmer's economy. The effects of diseases on plants are becoming a challenging approach in terms of crop protection and production of healthy food. Traditional methods for the identification and diagnosis of plant diseases depend mainly on the visual analysis of an expert in the area, or a study in the laboratory. These studies generally require a high professional knowledge in the field, beside the probability of failure to successfully diagnose specific diseases, which consequently led to erroneous conclusions and treatments (Ferentinos, 2018). Under those circumstances, to obtain a fast and accurate decision, an automatic system would offer a highly efficient support to identify diseases and pest of infected plants (Mohanty et al., 2016; Fuentes et al., 2017b). Recent advances in computational technology, in particular, Graphics Processing Units (GPUs), have led to the development of new image-based technology, such as high efficient deep neural networks. The application of deep learning has been also extended to the area of precision agriculture, in that, it has shown a satisfactory performance when dealing with complex problems in real time. Some applications include the study of diseases identification of several crops, such as tomato (Fuentes et al., 2017b), apple (Liu et al., 2018), banana (Amara et al., 2017), wheat (Sankaran et al., 2010), cucumber (Kawasaki et al., 2015).

CNN-based methods constitute a powerful tool that has been used as a feature extractor in several works. Mohanty *et* al. (Mohanty et al., 2016) compare two CNN architectures AlexNet and GoogLeNet to identify 14 crop species and 26 diseases using a large database of diseases and healthy plants. Their results show a system that is able to efficiently classify images that contain a particular disease in a crop using transfer learning. However, the drawback of this work is that its analysis is only based on images that are collected in the laboratory, not in the real field scenario. Therefore, it does not cover all the variations included there. Similarly, Sladojevic et al. (2016) identify 13 types of plant diseases out of healthy leaves with an AlexNet CNN architecture. They used several strategies to avoid overfitting and improve classification accuracy, such as data augmentation techniques to increase the dataset size, and finetuning to increase efficiency while training the CNN. The system achieved an average accuracy of 96.3%. Recently, Liu et al. (2018) proposed an approach for apple leaf disease identification based on a combination of AlexNet and GoogLeNet architectures. Using a dataset of images collected in the laboratory, that system is trained to identify four types of apple leaf diseases with an overall accuracy of 97.62%. In (Ferentinos, 2018), Ferentinos evaluates various CNN models to detect and diagnose plant diseases using leaves images of healthy and infected plants. The system is able to classify 58 distinct plant/disease combinations from 25 different plants. In addition, the experimental results show an interesting comparison when using images collected in the laboratory vs. images collected in the field. Promising results are presented using both types of images, with the best accuracy of 99.53% given by a VGG network. However, the success rate is significantly lower when images collected in the field are used for testing instead of laboratory images. In fact, according to the author, this demonstrates that image classification under real field conditions is much more difficult and complex than using images collected in the laboratory.

Although the works mentioned above show promising results in the task of plant diseases identification, challenges such as the complex field conditions, variation of infection, various pathologies in the same image, surrounding objects, are not investigated. They mainly use images collected in the laboratory, and therefore, do not deal with all the conditions presented in a real scenario. Furthermore, they are diseases classification-based methods.

In contrast, Fuentes et al. (2017b) presented a system that is able to successfully detect and localize 9 types of diseases and pests of tomato plant using images collected in the field, including real cultivation conditions. That approach differs from the others in that it generates a set of bounding boxes that contain the location, size, and class of diseases and/or pest in the image. This work investigates different meta-architectures and CNN feature extractors to recognize and localize the suspicious areas in the image. As a result, the authors show a satisfactory performance of 83%. However, the system presents some difficulties that do not allow it to obtain a higher performance. They mention that due to the lack of samples, some classes with high variability tend to be confused with others, resulting in false positives or lower precision.

Following the idea in (Fuentes et al., 2017b), our current work aims to address the problems mentioned above and improve their results by focusing on false positives and class unbalance issues. On the other hand, our approach studies several techniques to make the system more robust against the inter- and intra-class variations of tomato diseases and pests.

The Problem of False Positives

Although the efficiency of object detectors has been improved since deeper neural networks are used as feature extractors, they cannot be generalized for all applications. In addition to the complexity of collecting a dataset for a specific purpose, class unbalance has shown to be a problem when training deep networks for object detection. Consequently, the number of false positives generated by the network is high, which in fact results in a lower precision rate.

In classification problems, the error can be caused by many facts. It can be a measure of true positives (correct classification) and true negatives compared to false positives (false alarms) and false negatives (misses). In object detection, the false positives deserve special attention as they are used to calculate precision. A higher number of false positives yields a lower precision value. Therefore, several techniques have been proposed to overcome this issue. For instance, in (Sun et al., 2016), the problem of object classification and localization is addressed by Cascade Neural Networks that use a multi-stream multi-scale architecture without object-level annotations. In this work, a multi-scale network is trained to propose boxes that likely contain objects, and then a cascade architecture is constructed by zooming onto promising boxes and train new classifiers to verify them. Another approach in (Yang et al., 2016), proposes a technique based on the concept of divide and conquer. Each task is divided via cascade structure for proposal generation and object classification. In proposal generation, they add another CNN classifier to distinguish objects from the background given the output of a previous Region Proposal Network. In the classification task, a binary classifier for each category focuses on false positives caused by mainly inter- and intra-category variances.

Hard Examples Mining

In conventional methods, an important assumption to trade off the error generated by the high number of false positives is mentioned in (Viola and Jones, 2001). They suggest that setting a threshold yields classifiers with fewer positives and lower detection rate. Lower thresholds yield classifiers with more false positives and higher detection rate. However, at this point, that concept is unknot yet clear, whether adjusting a threshold preserves the training and helps generalization in deep learning.

Recently, the concept of hard examples mining has been applied to make the training of neural networks easier and efficient. In (Shrivastava et al., 2016), a technique called “Online Hard Example Mining” (OHEM) aims to improve the training of two-stage CNN detectors by constructing mini batches using high-loss examples. This technique removes the need for several heuristic and hyperparameters used in Region-based Convolutional Networks by focusing on the hard-negative examples. In contrast, the scope of this work is to understand whether the use of a refinement strategy can deal with the false positives generated by an object detection network.

The design of our multi-level approach points out two steps for object detection with a specific application in tomato diseases and pest recognition, in particular, the concept of Region-Based Neural Networks for bounding box generation (Fuentes et al., 2017b) and the CNN filter bank for "false positives" reduction. We emphasize that although our previous approach (Fuentes et al., 2017b) shows a satisfactory performance, the results can be further improved with the techniques proposed in our current approach. This aims to make the system more robust to inter- and intra-class variations.

DIAGNOSIS SYSTEM WITH REFINEMENT FILTER BANK

System Overview

Our approach proposes a method to detect diseases and pests of tomato plants using technology based on Deep Learning. The system consists of three basic components: a primary diagnosis unit (Bounding Box Generator), a secondary diagnosis unit (CNN filter bank), and an integration unit. For each image and class category, the primary unit generates a set of bounding boxes with scores of a specific class instance, and the coordinates that indicate the location of the target. Then, the secondary unit filters the confidence of each box by training CNN classifiers independently for each class to further verify their instance. Finally, the integration unit combines the results from the primary and secondary units. **Figure 1** illustrates the overall proposed system.

Primary Diagnosis Unit

We follow the system proposed in (Fuentes et al., 2017b) that implements a meta-architecture and several feature extractors to handle detection and recognition of complex diseases and pests in images. The input of the system is an image of any arbitrary size. In the first part of the framework, the primary diagnosis unit (bounding box generation) proposes a set of boxes that contain the suspicious areas of the image. That is, for an input image I and 10 object categories $C = \{1, 2, 3, \ldots, 10\}$, we want to extract the object proposals

$$\mathbf{b_i} = \{\mathbf{s_i},\ \mathbf{l_i},\ \mathbf{c_i}\},\ \mathbf{i} = \mathbf{1, 2, \ldots, B_I} \tag{1}$$

where B_I is the number of bounding boxes detected from the image I, and b_i is the ith bounding box. The set of bounding boxes provide information such as the size s, location l and class score c.

The following sub-sections show the main characteristics of the primary diagnosis unit.

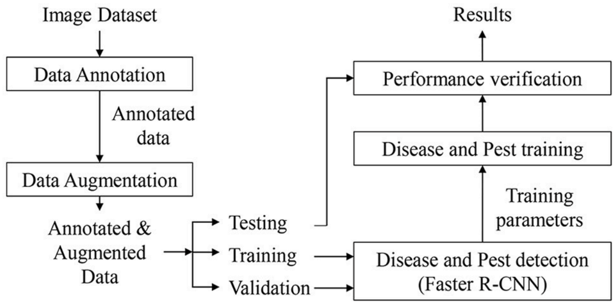


FIGURE 2 | Primary Diagnosis Unit for bounding box detection. Similar to Fuentes et al. (2017b).

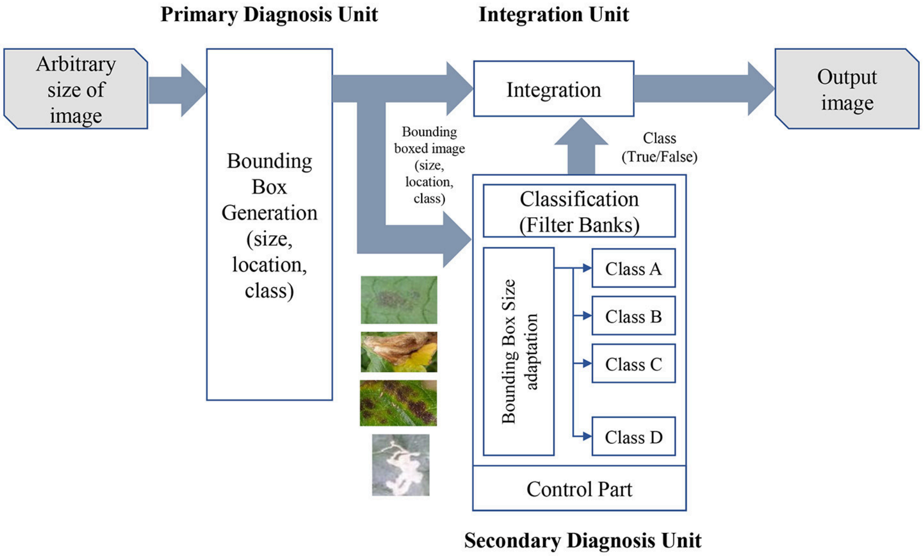


FIGURE 1 | A general overview of our proposed approach. The input images with an arbitrary size are trained in our primary diagnosis unit that generates bounding boxes along with their location and class of the infected areas in the image. The set of bounding boxes is used as input in the secondary diagnosis unit, which independently trains CNN filter banks for each class, with the purpose of reducing the number of false positives generated by the primary unit. Both systems are further integrated into class and location.

Faster R-CNN for Bounding Box Detection

Figure 2 shows the process followed by the primary diagnosis unit to detect the suspicious areas containing diseases and pests in the input image. This part is mainly based on the Faster R-CNN. It uses a Region Proposal Network (RPN) to generate a feature map through a CNN and proposes vectors by convolving them using a sliding-window method. The size, location, and class score (probability of having an object or not) are generated for each bounding box proposed by the network. Finally, the object detection is completed by applying Fully-Connected layers to classify the obtained bounding boxes called Regions of Interest (ROIs). **Figure 3** shows a representation of some bounding boxes that contain suspicious areas obtained through the primary diagnosis unit.

False Positives Identification

The performance of the system is evaluated as the average precision (AP) introduced by the Pascal VOC Challenge. The AP is the area under the Precision-Recall curve for detection. It has a constant interval Recall level [0, 0.1, ..., 1], and is the mean AP calculated for all classes, as shown in Equations (2, 3).

$$AP = \frac{1}{11} \sum_{r \in \{0,0.1,\ldots,1\}} P_{\mathrm{int}\,erp}(r) \tag{2}$$

$$P_{\mathrm{interp}}(r) = \max_{\tilde{r}\,:\,\tilde{r} \geq r} p(\tilde{r}) \tag{3}$$

where, $P_{interp}\ (r)$ is the maximum precision for any recall values greater than r, and $p(\tilde{r})$ is the measured precision at recall $\tilde{r}$. Then the AP is computed as the average of $P_{interp}\ (r)$ at all recall levels. IoU, defined in Equation 4, is a widely-used metric for evaluating the accuracy of object detectors.

$$IoU(A, B) = \left| \frac{A \cap B}{A \cup B} \right| \tag{4}$$

where A represents the ground-truth box collected in the Annotation and B represents the prediction result of the network. If the estimated IoU is higher than the threshold, the predicted results will be considered as positive samples (TP + FP), otherwise as negatives (FN + TN). TP, FP, FN, and TN represent the True Positives, False Positives, False Negatives and True Negatives respectively. Ideally, the number of FPs and FNs should be small and the network must determine how accurately each case can be handled.

Table 1 shows the number of True Positive and False Positive bounding boxes generated by the primary diagnosis unit for each class using the Faster R-CNN detector when the IoU threshold = 0.5. The results evidence a relation of 89.97% TP and 10.03% FP of the total of bounding boxes generated.

The IoU is a parameter that is used to determine whether a detected bounding box is a TP, TN, FP, or FN. However, the number of false positives may vary for each class, due to in part to the complexity and number of samples available. Additionally, they represent a problem mainly caused by the inter- and intra-class variations presented in the dataset. To determine this relationship, we extract the bounding boxes from the primary diagnosis unit and evaluate the detection results with different IoU thresholds. As shown in **Figure 4**, we notice an unbalance between the positive classes (diseases and pest) and the background class (negative class) is highly visible. In fact, since the number of examples for some classes such as leaf mold and yellow curl virus is relatively high compared to other classes. Consequently, the system tends to give higher priority to cases with a greater source of information.

Notably, it is common that the number of positive samples detected by the primary diagnosis unit decreases as the IoU threshold is increased. However, the impact on the recall should be also considered in terms of the number of false negatives.

TABLE 1 | Identification of True positive and false positive bounding boxes generated by the primary diagnosis unit.

Class	True positives	False positives	Total
Leaf mold	11022	900	11922
Gray mold	1642	1126	2768
Canker	2226	422	2648
Plague	2246	324	2570
Miner	5198	85	5283
Low temperature	426	51	477
Powdery mildew	314	24	338
Whitefly	380	24	404
Yellow leaf curl	3819	108	3927
Nutritional excess	403	23	426
Total TP	27676 (**89.97%**)*		
Total FP		3087 (**10.03%**)*	
Total Samples			30763

The percentage value corresponds the portion respect to the total.

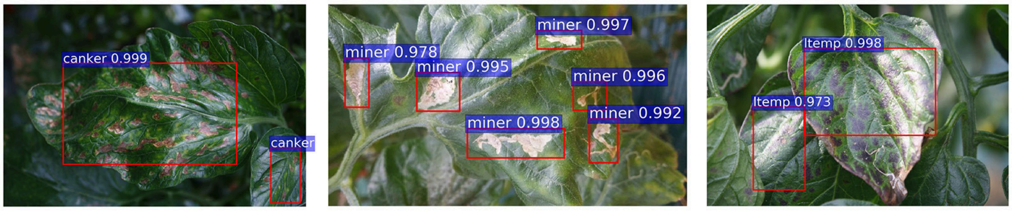


FIGURE 3 | A representation of bounding boxes with various sizes for different detected classes.

Since our training data have a large unbalance between classes, we investigate whether changing the IoU threshold value produced any change in the number of positive samples. **Figure 4** shows the portion of bounding boxes detected by the primary diagnosis unit with respect to IoUs and classes. The unbalance between the classes from the original training data is not reduced but it becomes even larger as the IoU threshold increases. When one of the target classes contains a much smaller amount of training data than the other target classes, it may be dominated by the others. Especially, if the class has a relatively large intra-class variation and small inter-class variation, its performance will be further degraded. In that case, the detector will produce more false positives for that class and the other classes as well.

Figure 5 shows some examples of false positives generated by the primary diagnosis unit. We present cases of canker, gray mold, and low temperature samples that have been misclassified as plague, canker, and canker, respectively. To improve the performance of the entire system, we need to investigate a strategy that allows the system to keep the true positives while handling the false positives.

As can be seen in **Figure 6,** due to the limited data available, the unbalance between classes results in lower performance. Each representation in **Figure 6** shows the precision-recall curves of the primary diagnosis unit using different IoU threshold values from 0.1 to 0.9. The precision-recall curves of the primary diagnosis unit illustrate that classes with more samples tend to be more stable and, therefore they may obtain a higher score. In addition, as the IoU value is increased, the performance of the system decreases and, consequently, some classes tend to be more affected since they may get confused among themselves or with others. This could be the case when more than one pathology is found in the sample area of the plant or is a consequence of various infection status with different visible patterns. Furthermore, we might also argue that there should be a tradeoff between the precision and recall when choosing a proper threshold value for the evaluation.

To visualize the individual performance of each class, we evaluate the average precision at different IoU threshold values. **Figure 7** shows that some classes such as leaf mold, canker, plague, yellow curl virus, nutritional excess show even better performance than the mean average precision. However, some critical classes like powdery mildew and miner experience worse performance as the IoU value is increased. These classes represent the challenging pathologies that may cause several detection inconveniences in the primary diagnosis unit.

In order to address the problem of false positives and improve the detector stability and performance of the system, we introduce the secondary diagnosis unit. To achieve that purpose,

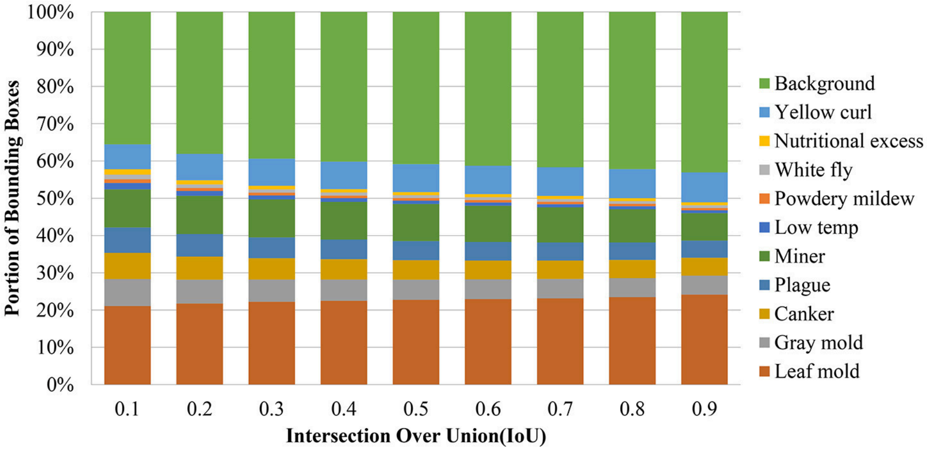


FIGURE 4 | The result of the bounding box detector evidence an unbalance between classes. Each column represents a comparison of the number of bounding boxes of each class using different intersection over union thresholds from 10 to 90%.

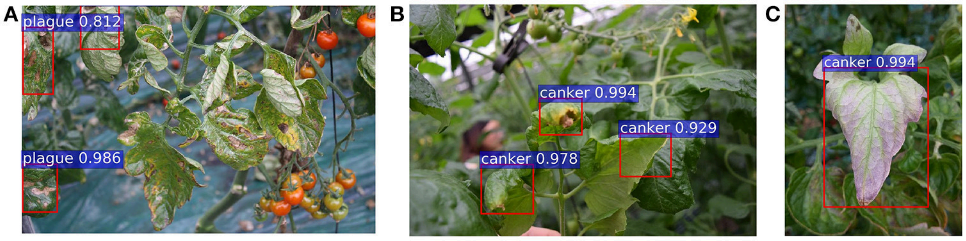


FIGURE 5 | A representation of some false positives generated in the primary diagnosis unit: **(A)** canker samples are detected as plague; **(B)** gray mold samples are detected as canker; **(C)** a low temperature sample is detected as canker.

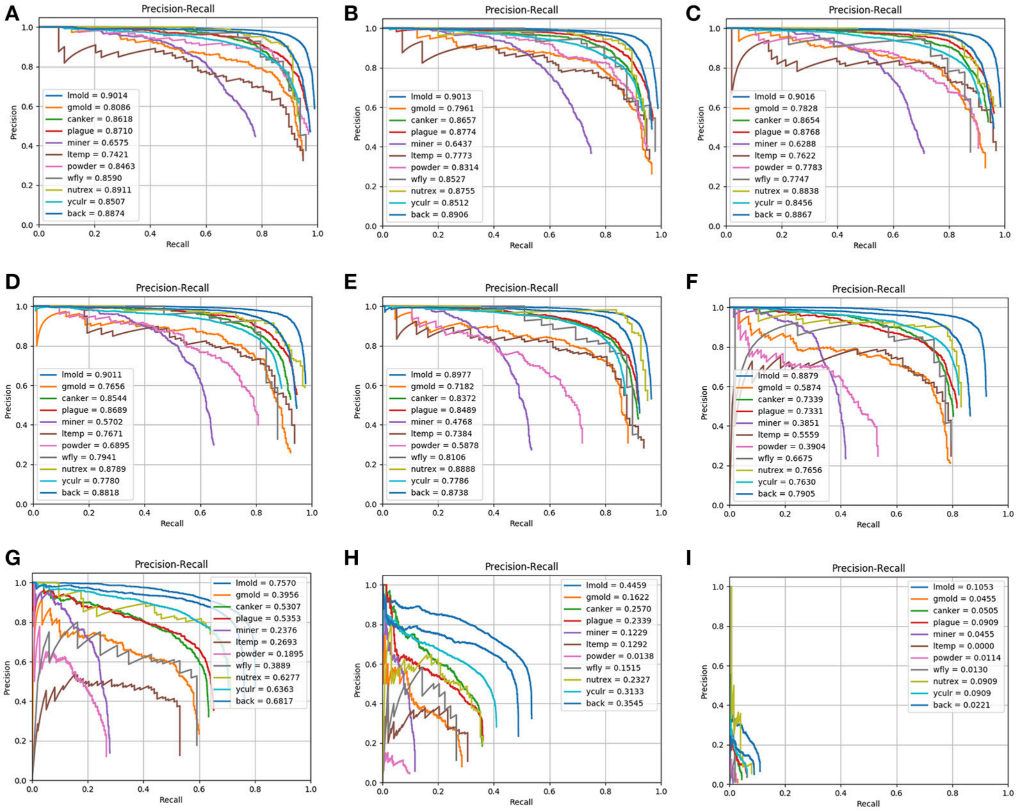


FIGURE 6 | Precision-Recall curves of the primary diagnosis unit (bounding box generation) for different IoU threshold values: **(A)** 0.1; **(B)** 0.2; **(C)** 0.3; **(D)** 0.4; **(E)** 0.5; **(F)** 0.6; **(G)** 0.7; **(H)** 0.8; **(I)** 0.9. Note that the performance decreases as the IoU value is increased.

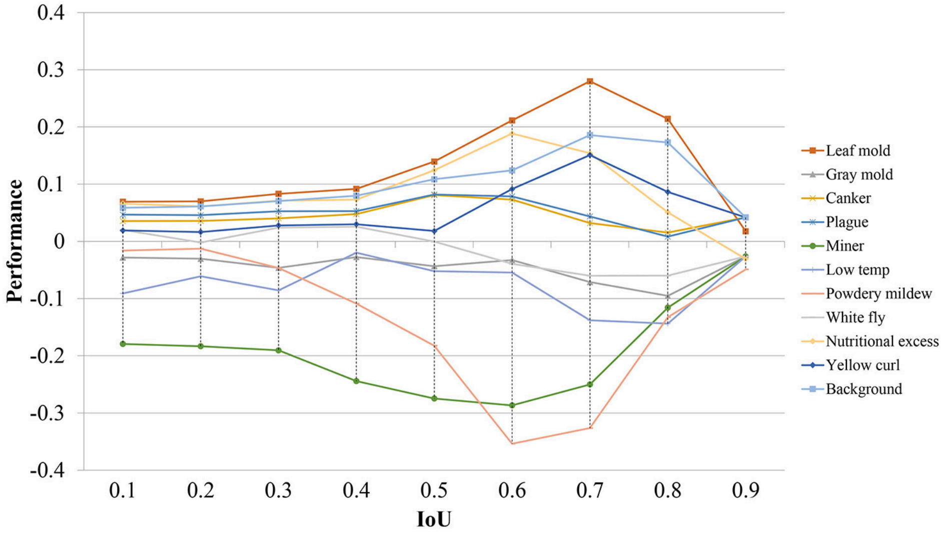


FIGURE 7 | Performance differences of all detected classes in terms of their average precision using different IoU threshold values. Note that some classes experience a positive performance, while others show a negative value.

this unit firstly sets the recall value $\mathbf{R}= \frac{\mathbf{TP}}{\mathbf{TP+FN}}$ and aims to improve the precision value $\mathbf{P}= \frac{\mathbf{TP}}{\mathbf{TP+FP}}$ using the filter bank. (The details of the filter bank are described in the next section).

Secondary Diagnosis Unit

The generated bounding boxes are very diverse in size and may contain different pathologies. Thus, the set of boxes are extracted and each one adjusted to an appropriate size before training the CNN filter bank. Within the classification block, there is a size adaptation that processes bounding boxes of various sizes and a control block that transfers data to the filter bank based on the information of the previously detected classes. **Figure 8** illustrates a general overview of the CNN filter bank.

Input Data

In this stage, regions that contain bounding boxes generated in the primary diagnosis unit by the Faster R-CNN are firstly extracted from the original images and then consecutively used by the CNN filter bank. They are divided into 10 different types of diseases and pests. Additionally, we include an extra class called "background." This class basically contains healthy areas of the plant or parts of the main scenario. **Figure 9** shows some examples of the bounding boxes used as input of the filter banks.

The number of classification blocks depends on the number of classes to be diagnosed. In addition, another function of the control block is to perform a process of adapting the size of the bounding boxes, before entering their respective CNN classifier. Each CNN determines either True or False values by estimating the probability of a disease or pests that appear in the input image.

Filter Bank Architecture

To address the problem of false positives caused by misclassification, we propose to use the secondary diagnosis unit that includes a CNN filter bank for each category. The added classifier plays a role of a judge that decides whether a bounding box is likely containing the correct target or not. In the CNN filter bank, each CNN directs a specific proposal to a particular object category, which in fact, also includes false positives as negative samples to make the system more robust against intra- and inter-class variations. The characteristics of the filter bank are introduced below.

a) Scale Adaptation

We construct a filter bank which contains *k-CNNs*, where *k* is the number of classes. Every CNN is an independent network but with the same number of parameters. Given a set of bounding boxes for each category, the control block first adapts the sizes of the bounding boxes to two scales: small and large, and feeds them into their respective CNN. To facilitate the process, each box is sampled to 300 × 300 and 500 × 500.

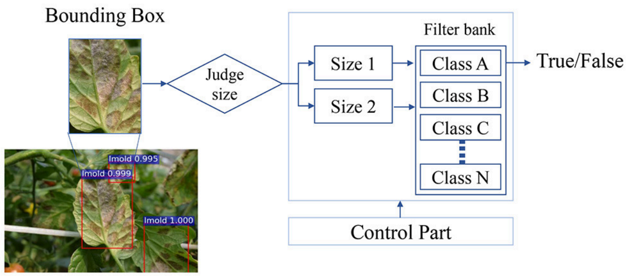


FIGURE 8 | A representation of a CNN filter bank for one class. The input images of the filter bank are the bounding boxes generated in the primary diagnosis unit. A judge step establishes the size of the image prior to its entrance to the CNN. The result is a decision of whether a target belongs to the category as it was detected (True) or not (False).

b) Filter Bank

Our *k*-CNNs are implemented in Caffe. For each network, we use a simple CNN architecture with 5 convolutional layers and 3 fully-connected layers. **Figure 10** illustrates a representation of a CNN architecture used in the Filter Bank.

To deal with the problem of false positives caused by misclassification, we consider our filter bank-based approach as an additional classifier for each object category. We find it important to train each CNN independently using the detection output (bounding boxes) of a specific category, so the detection should have a higher score on that category. To that effect, each CNN uses bounding boxes specific to one category, which allows to capture intra-class variation.

During the training process, first, the primary diagnosis unit (bounding box generator) is trained on the training dataset. Then, the bounding boxes (set of true positive, false positive and true negative boxes) obtained from the primary diagnosis unit are used to train the secondary diagnosis unit (filter banks). Further, the set of boxes containing the true targets are selected as positives samples and, the false positives along with true negative samples (hard negatives) are used as negative samples. The proposed approach works like a filter whose goal is to preserve bounding boxes with higher recognition rate while eliminating the false positives and negatives from the list. As shown in **Figure 10**, a CNN structure for class diagnosis is examined, and the final result is a precision value of a specific class performed by a single CNN network. To make the training process effective, both units are trained and optimized consecutively with shared convolution weights.

During testing, using an input image, the primary diagnosis unit generated a set of bounding boxes that contain the object categories. Then, each detection is again classified by the secondary diagnosis unit. As both units share weights, the image feature maps are computed only once during testing.

The advantage of this structure is that it can respond effectively to diseases or pests that appear in the images. Basically, the system consists of a modular architecture that can be adapted to as many categories as required. It is also possible to include more categories simply by adding a CNN to the filter bank.

Improving the Precision Results

The purpose of this technique is to increase the precision score. This is a technique commonly used for object detection, but has been adapted for our application. Therefore, increasing the precision score is the most important factor in measuring the efficiency of this technology.

Figure 11 shows a representation of images of tomato plants used for learning. The yellow rectangle represents the suspicious areas of the disease or pests located in the foreground, and the rest is considered the background. The areas annotated within the yellow bounding boxes are considered positive samples of their respective class, and the False Positive or True Negative samples are selected either as part of another class and as background

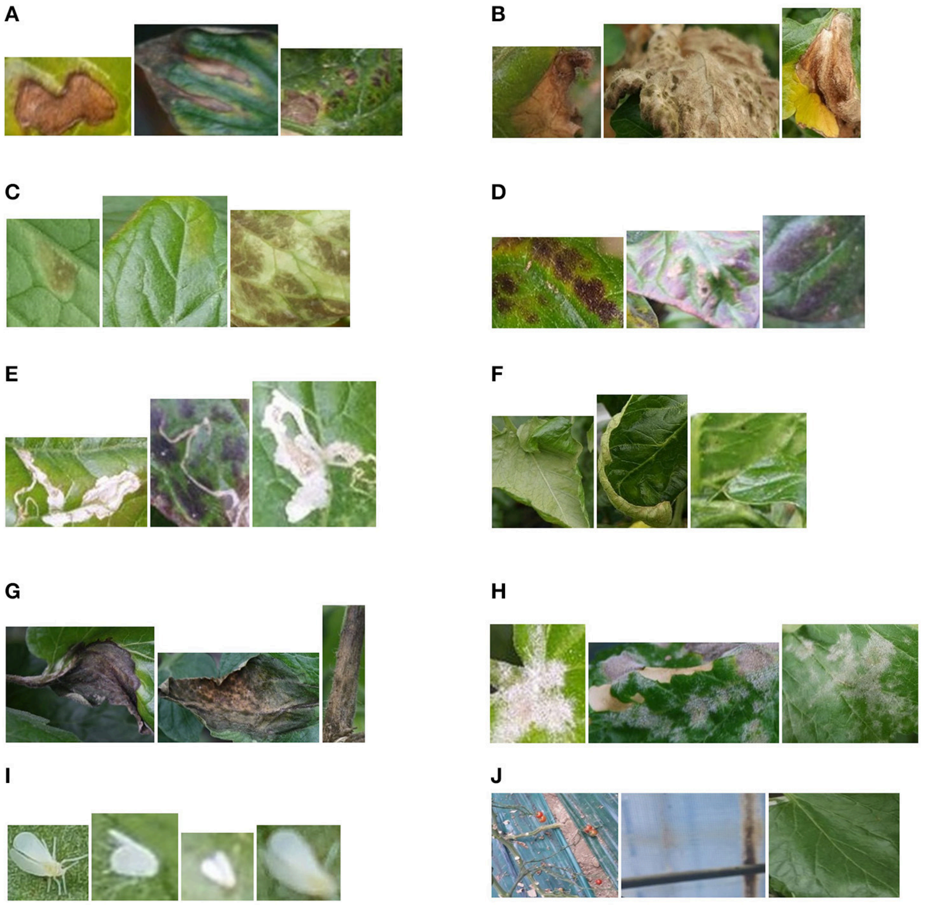


FIGURE 9 | Areas containing suspicious infections due to diseases and pests that are generated by the primary diagnosis unit and used as input to the CNN Filter Banks. **(A)** Canker, **(B)** Gray mold, **(C)** Leaf mold, **(D)** Low temperature, **(E)** Miner, **(F)** Nutritional excess, **(G)** Plague, **(H)** Powdery mildew, **(I)** Whitefly, **(J)** Background.

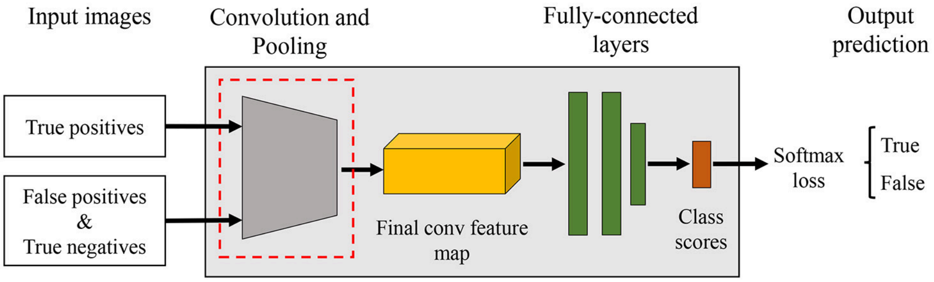


FIGURE 10 | Example of a CNN architecture used in the filter bank. The goal of each CNN is to verify if an input bounding box is likely containing the target category or not, as well as, to make the system more robust against intra- and inter-class variations.

of the image. Nevertheless, it is necessary to emphasize that all samples containing the suspicious areas should be annotated, to avoid confusing the system when testing in unseen images.

Complexity of the CNN Filter Bank

The application of a secondary diagnosis unit (CNN Filter Banks) allows the system to achieve higher precision while maintaining a reasonable computational cost. The CNN models of the filter banks are a flexible framework with different design selections. We make several modifications to the architecture to verify the performance of the network. Therefore, we have extended the design to understand the number of layers needed for the system to be accurate enough. Through this technique, we can find a suitable solution for our application without sacrificing system performance. **Figure 12** shows different CNN architectures that are further tested in the experimental results.

They consist of 5, 4, 3, 2, and 1 convolutional layers respectively. Each convolutional network represents the area within the red bounding box in **Figure 12**.

Integration Unit

The integration unit (see **Figure 1**) combines the results from the primary diagnosis unit (bounding box generation) and the secondary diagnosis unit (CNN filter bank). The result of the CNN filter bank is a decision on whether the target corresponds to the category as it was detected (True) or not (False). Next, the integration unit has two main functions: (1) it combines the information of primary and secondary units, and (2) it keeps the True Positive samples and eliminates the False Positives that were misclassified in the first stage. As mentioned earlier, a smaller number of False Positives helps to improve the precision score. The whole process operates autonomously, which allows the system to provide accurate results in real time.

EXPERIMENTAL RESULTS

Tomato Diseases and Pests Dataset

We conduct experiments on our Tomato Diseases and Pest Dataset, as in (Fuentes et al., 2017b). This dataset consists of approximately 5,000 images collected from several tomato farms located in different areas of South Korea. Diseases and Pest can be developed under different conditions such as climate, location, humidity, etc. Therefore, using simple camera devices, the images were collected in various conditions depending on the time (e.g., illumination), the season (e.g., temperature, humidity), and the place where they were taken (e.g., greenhouse) (Fuentes et al., 2017b). Additionally, our dataset includes images with various resolutions, samples in the early, middle, and last infection status, images containing different infected areas in the plant (e.g., stem, leaves, fruits, etc.), different sizes of plants, objects that surround the plant in the greenhouse, etc. The categories and number of samples for each class are presented in **Table 2.** The number of annotated samples corresponds to the bounding boxes annotated in the images after applying the following data augmentation techniques: geometric transformations (resizing, crop, rotation, horizontal flipping) and intensity transformations (contrast and brightness enhancement, color, noise). The background class is a transverse class that has been annotated in most of the images, and its bounding boxes are used as negative samples during training the CNN filter bank.

In addition to the dataset used in (Fuentes et al., 2017b), we have also included a new class that contains images of the "yellow

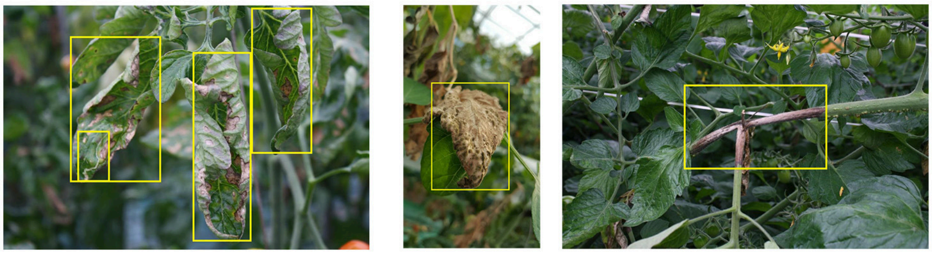


FIGURE 11 | A representation of various images with bounding boxes used for training. The yellow boxes represents the suspicious infected areas of the plant.

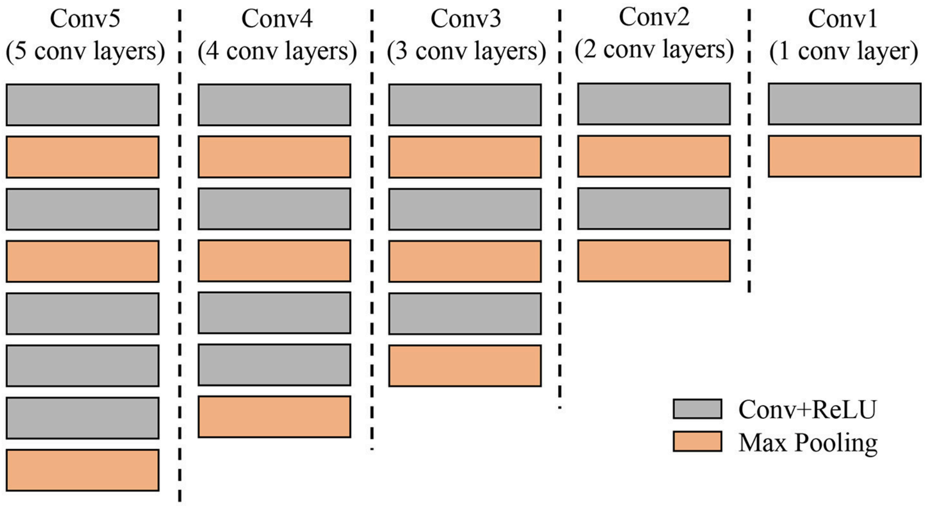


FIGURE 12 | Different designs of Convolutional Neutral Networks used in the experiments to understand the required parameters of the secondary diagnosis unit. Each column represents a single CNN from 5 to 1 convolutional layers (gray color).

leaf curl" virus. As mentioned earlier, we have identified that one of the main difficulties that limit the system to obtain higher precision is the unbalance between classes due to the conditions and limited data available. This can be evidenced by the number of images that belong to each class, as shown in **Table 2** and **Figure 4**.

TABLE 2 | List of categories included in our tomato diseases and pests dataset and their annotated samples.

Class	Number of Images in the Dataset[a]	Number of Annotated Samples (Bounding Boxes)[b]	Percentage of Bounding Box Samples (%)
Leaf mold	1,350	11,922	24.06
Gray mold	335	2,768	5.57
Canker	309	2,648	5.33
Plague	296	2,570	5.17
Miner	339	5,283	10.63
Low temperature	55	477	0.96
Powdery mildew	40	338	0.68
Whitefly	49	404	0.81
Nutritional excess	50	426	0.85
Yellow leaf curl	3,927	3,927	7.90
Background[c]	2,177	18,899	38.03
Total	8,927	49,662	100

[a] *Number of images in the dataset.*

[b] *Number of annotated samples after data augmentation.*

[c] *Transverse category included in every image.*

Experimental Setup

Our proposed system has been trained and tested on two NVidia GeForce TitanXP GPUs. We conducted experiments on our Tomato Diseases and Pest dataset, using an extensive data augmentation to avoid overfitting. The data has been distributed as follows:

- Primary diagnosis unit: from the whole number of images in the dataset, 80% are used for training, 10% for test and the remaining 10% for validation.
- Secondary diagnosis unit: depending on the number of True Positives and False Positives mentioned in **Table 1**, we divide them into 80% for training and 20% for test. However, since the number of images for some classes is limited, we include samples from other classes as negative samples in each CNN to avoid problems of class unbalance during training and test.

Complexity of the CNN Filter Bank

We are interested in observing how the performance changes in different levels of the Convolutional Neural Network. For this purpose, we have trained a set of CNNs with a various number of layers in the filter bank. We found that models with fewer layers are more likely to be overfitted. Since the amount of data is still small for some classes, we also found that although CNNs with one and two layers learn during training, they are not able to generalize well during testing. CNNs with three or four layers show acceptable performance, but a CNN with 5 layers tends to be more stable during testing. The results can be seen in **Figure 13**.

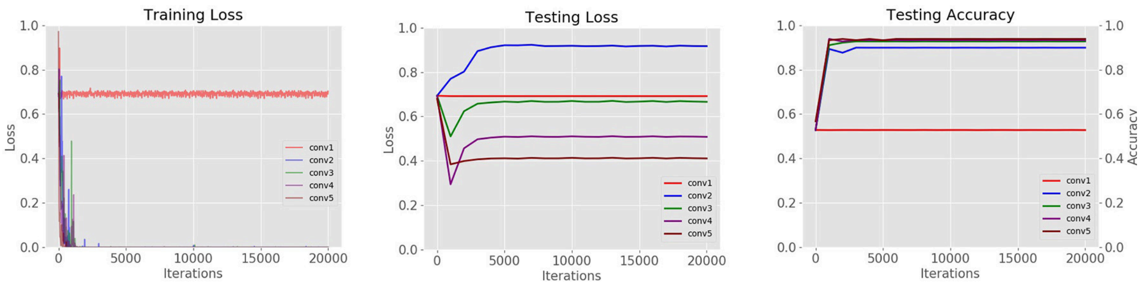


FIGURE 13 | Loss curves of CNN architectures with different layers studied in the refinement filter bank of our proposed approach.

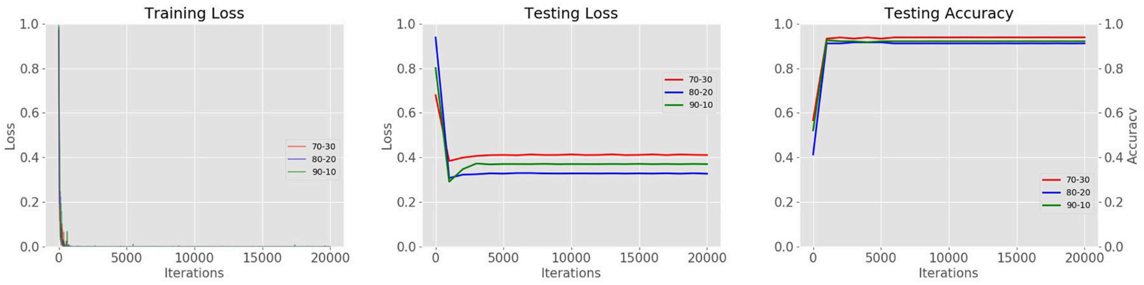


FIGURE 14 | Results of data distribution for cross-validation. Using three different combinations of data (70–30%, 80–20%, 90–10% trainval and testing respectively).

Data Distribution

Using different combinations of training-validation (trainval) and testing data, we are also able to find the best combination that allows the system to generate better results. The goal is not only to improve the precision value, but also to propose a system that is stable during training and testing. Therefore, we trained and tested the system with different combinations of data. We found that a distribution of 80% training and 20% testing shows more stability throughout the iterations in the testing loss curve, in contrast to the results of the testing accuracy curve where combination a 70% training, 30% testing shows better performance. The results of data distribution with different combinations are illustrated in **Figure 14**.

Quantitative Results

Table 3 shows the final results of our refinement system. The comparative values evidence a satisfactory improvement in all classes with respect to our previous results (Fuentes et al., 2017b). The mean Average Precision demonstrates an improvement of about 13%. This is, in fact, due to the implementation of the secondary diagnosis unit (CNN filter bank) that allows the system to filter misclassified samples focusing mainly on the reduction of false positives.

TABLE 3 | Comparative results of our proposed approach with the previous system (Fuentes et al., 2017b).

Class	FRCNN (VGG-16)	Refinement Filter Bank (Proposed)	Difference of accuracy
Leaf mold	0.9060	0.9205	0.0145
Gray mold	0.7968	0.8910	0.0942
Canker	0.8569	0.9376	0.0807
Plague	0.8762	0.9710	0.0948
Miner	0.8046	0.9947	0.1901
Low temperature	0.7824	0.9821	0.1997
Powdery mildew	0.6556	0.9963	0.3407
Whitefly	0.8301	0.9929	0.1628
Nutritional excess	0.8971	0.9893	0.0922
Yellow leaf curl	0.8500	0.9500	0.1000
Mean AP	0.8255	0.9625	0.1370

The number of samples and variations are another key points that influence in the final results. For example, in the case of gray mold, the number of samples is smaller than leaf mold. Moreover, the gray mold class shows a high intra-class variability that could confuse the system with other classes (See **Figure 4**).

Does the CNN Filter Bank Help?

The input images of the filter bank are the set of bounding boxes generated by the Bounding Box Generator. The control part sets the size of the images before entering the CNN filter bank. The result is defined as "True" if the image falls into the same category as it was detected or "False" otherwise.

Figure 15 shows the Training Loss, Testing Loss, and Testing Accuracy of the CNN filter bank for the most challenging classes in the dataset such as leaf mold, plague, and canker.

Due to the implementation of a secondary diagnosis unit, the results have been substantially improved compared to the previous results reported in (Fuentes et al., 2017b). Therefore, we might argue the importance of the CNN filter bank toward reducing the number of false positives. As presented in **Table 3**, the mean Average Precision has been increased in approximately 13% compared to the best results generated by the Faster R-CNN with a VGG-16 feature extractor in (Fuentes et al., 2017b).

An additional benefit of using a second diagnosis unit is the easy configuration of the framework. The CNN Filter Bank that is composed by a set of CNN architecture, as the one shown in **Figure 10**. This modular architecture is able to add another network if the study requires including more classes by changing the structure shown in **Figure 8**.

CONCLUSION AND FUTURE WORKS

In this work, we have proposed a framework based on deep neural networks that performs onto promising object-specific bounding boxes for efficient real time recognition of diseases and pests of tomato plants. Our detector uses images captured in the field by various camera devices and process them in real time. The detector is composed of three units: A primary diagnosis unit (bounding box generator) first learns to propose bounding boxes with size, location, and class through a Region-based Neural Network trained with the input images. The promising bounding boxes that belong to each class are then used as

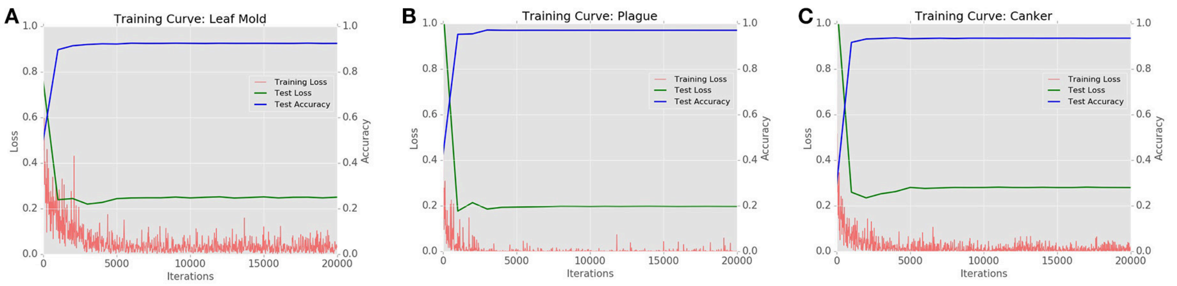


FIGURE 15 | Training curves generated by the CNN filter bank for the most challenging classes of the system **(A)** Leaf mold with AP: 92%, **(B)** Plague with AP: 97%, **(C)** Canker with AP: 93%.

input to the secondary diagnosis unit (CNN filter bank) for verification. This secondary unit filters misclassified samples by training independent CNN classifiers for each class. The result of the CNN Filter Bank is a decision on whether the target corresponds to the category as it was detected (True) or not (False) otherwise. Finally, an integration unit combines the information from the primary and secondary units by keeping True Positive samples and eliminating False Positives that were wrongly misclassified in the first stage. By this implementation, the proposed approach outperforms our previous results by a margin of 13% mean Average Precision in the task of tomato diseases and pest recognition. Furthermore, our system is able to deal with the problems of false positives generated by the Bounding Box Generator, and class unbalances that appear especially in datasets with limited data. We expect that our work will significantly contribute to the agricultural research area. Future works will focus on extending our approach to other types of crops.

AUTHOR CONTRIBUTIONS

AF designed the study, performed the experiments, and data analysis, and wrote the paper. DP and SY advised on the design of the system and analyzed to find the best method for efficient recognition of diseases and pests of tomato plants. JL provided the facilities for data collection and contributed with the information for the data annotation.

FUNDING

This work was supported by the Brain Korea 21 PLUS Project, National Research Foundation of Korea. This work was carried out with the support of Cooperative Research Program for Agriculture Science and Technology Development (Project No. PJ0138912018) Rural Development Administration, Republic of Korea.

REFERENCES

Amara, J., Bouaziz, B., and Algergawy, A. (2017). "A deep learning-based approach for banana leaf diseases classification," in *BTW 2017 - Workshopband, Lecture Notes in Informatics (LNI), Gesellschaft für Informatik* (Bonn), 79–88.

Bloice, M., Stocker, C., and Holzinger, A. (2017). Augmentor: an image augmentation library for machine learning. *arXiv:1708.04680*.

Chen, Y., Li, J., Xiao, H., Jin, X., Yan, S., and Feng, J. (2017). "Dual Path Networks," in *IEEE Conference on Computer Vision and Pattern Recognition* (Honolulu).

Dai, J., Li, Y., He, K., and Sun, J. (2016). "R-FCN: object detection via region-based fully convolutional networks," in *IEEE Conference on Computer Vision and Pattern Recognition* (Las Vegas, NV).

Donatelli, M., Magarey, R. D., Bregaglio, S., Willocquet, L., Whish, J. P. M., and Savary, S. (2017). Modeling the impacts of pests and diseases on agricultural systems. *Agricult. Syst*. 155, 213–224. doi: 10.1016/j.agsy.2017.01.019

Ferentinos, K. (2018). Deep learning models for plant disease detection and diagnosis. *Comp. Electr. Agricut*. 145, 311–318. doi: 10.1016/j.compag.2018.01.009

Fuentes, A., Im, D. H., Yoon, S., and Park, D. S. (2017a). "Spectral analysis of CNN for tomato disease identification," in *Artificial Intelligence and Soft Computing, Lecture Notes in Computer Science, ICAISC 2017*, eds L. Rutkowski, M. Korytkowski, R. Scherer, R. Tadeusiewicz, L. Zadeh, and J. Zurada (Cham, Springer), 40–51.

Fuentes, A., Yoon, S., Kim, S. C., and Park, D. S. (2017b). A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. *Sensors* 17:E2022. doi: 10.3390/s17092022

Fuentes, A., Youngki, H., Lee, Y., Yoon, S., and Park, D. S. (2016). "Characteristics of tomato diseases–A study for tomato plant disease identification," in *Proceedings International Symposium on Information Technology Convergence* (Shanghai, China).

He, K., Zhang, X., Ren, S., and Sun, J. (2016). "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition* (Las Vegas, NV).

Hu, J., Shen, L., and Sun, G. (2017). "Squeeze-and-Excitation Networks," in *IEEE Conference on Computer Vision and Pattern Recognition* (Honolulu).

Huang, G., Liu, Z., Van der Maaten, L., and Weinberger, K. (2017). "Densely connected convolutional networks," in *IEEE on Computer Vision and Pattern Recognition* (Honolulu).

Ioffe, S., and Szegedy, C. (2015). Batch normalization: accelerating deep network training by reducing internal covariate shift. *arXiv:1502.03167.*

Johannes, A., Picon, A., Alvarez-Gila, A., Echazarra, J., Rodriguez-Vaamonde, E., Diez, A., et al. (2017). Automatic plant disease diagnosis using mobile capture devices applied on a wheat use case. *Comput. Electr. Agricult*. 138, 200–209. doi: 10.1016/j.compag.2017.04.013

Kawasaki, Y., Uga, H., Kagiwada, S., and Iyatomi, H. (2015). "Basic study of automated diagnosis of viral plant diseases using convolutional neural networks," in *Advances in Visual Computing. Lecture Notes in Computer Science,* Vol. 9475, eds G. Bebis et al. (Cham, Springer), 638–645.

Krizhevsky, A., Sutskever, I., and Hinton, G. (2012). "Imagenet classification with deep convolutional neural networks," in *NIPS'12 Proceedings of the 25th International Conference on Neural Information Processing Systems* (Lake Tahoe, NV).

Liu, B., Zhang, Y., He, D., and Li, Y. (2018). Identification of apple leaf diseases based on deep convolutional neural networks. *Sensors* 10:11. doi: 10.3390/sym10010011

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C., et al. (2016). "SSD: Single Shot MultiBox Detector," in *European Conference on Computer Vision – ECCV* (Amsterdam).

Mahlein, A. K. (2016). Plant diseases detection by imaging sensors – parallels and specific demands for precision agriculture and plant Phenotyping. *Plant Dis.* 100, 241–251. doi: 10.1094/PDIS-03-15-0340-FE

Mohanty, S. P., Hughes, D. P., and Salathé, M. (2016). Using deep learning for image-based plant disease detection. *Front. Plant Sci*. 7:1419. doi: 10.3389/fpls.2016.01419

Pereyra, G., Tucker, G., Chorowski, J., Kaiser, L., and Hinton, G. (2017). "Regularizing neural networks by penalizing confident output distributions," in *International Conference on Learning Representations* (Toulon).

Pinstrup-Andersen, P. P. (2002). The future world food situation and the role of plant diseases. *Plant Health Instruct.* 22, 321–331. doi: 10.1094/PHI-I-2001-0425-01

Redmon, J., Divvala, S., Girschick, R., and Farhadi, A. (2015). "You only look once: unified, real-time object detection," in *IEEE Conference on Computer Vision and Pattern Recognition* (Boston, MA).

Redmon, J., and Farhadi, A. (2017). "YOLO9000: better, faster, stronger," in *IEEE Conference on Computer Vision and Pattern Recognition* (Honolulu).

Ren, S., He, K., Girschick, R., and Sun, J. (2016). Faster R-CNN: towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Machine Intell.* 39, 1137–1149. doi: 10.1109/TPAMI.2016.2577031

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., et al. (2015). Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.* 115, 211–252. doi: 10.1007/s11263-015-0816-y

Russakovsky, O., Lin, Y., Yu, K., and Li, F. F. (2012). "Object-centric spatial pooling for image classification," in *Computer Vision – ECCV 2012*, *Lecture Notes in*

Computer Science, Vol. 7573, eds A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid (Heidelberg: Springer), 1–15.

Sankaran, S., Mishra, A., Ehsani, R., and Davis, C. (2010). A review of advanced techniques for detecting plant diseases. *Comput. Electron. Agricult.* 72, 1–13. doi: 10.1016/j.compag.2010.02.007

Savary, S., Ficke, A., and Aubertot, J. N. (2012). Crop losses due to diseases and their implications for global food production losses and food security. *Food Sec.* 4, 519–537. doi: 10.1007/s12571-012-0200-5

Shrivastava, A., Gupta, A., and Girshick, R. (2016). Training region-based object detectors with online hard example mining. *arXiv:1604.03540*.

Simonyan, K., and Zissermann, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556*.

Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D., and Stefanovic, D. (2016). Deep neural networks based recognition of plant diseases by leaf image classification. *Comput. Intell. Neurosci*. 2016:3289801. doi: 10.1155/2016/3289801

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *J Machine Learn. Res.* 15, 1929–1958. Available online at: http://jmlr.org/papers/v15/srivastava14a.html

Sun, C., Paluri, M., C1ollobert, R., Nevatia, R., and Bourdev, L. (2016). "ProNet: learning to propose object-specific boxes for cascaded neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition* (Las Vegas, NV).

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., et al. (2015). "Going deeper with convolutions," in *IEEE Computer Vision and Pattern Recognition* (Boston, MA).

Szegedy, C., Toshev, A., and Erhan, D. (2013). Deep Neural Networks for Object Detection," in *NIPS 2013 Conference on Neural Information Processing Systems*.

Van-Laarhoven, T. (2017). L2 Regularization versus batch and weight normalization. *arXiv:1706.05350v051*.

Viola, P., and Jones, M. (2001). "Robust Real-time Object Detection," in *International Workshop on Statistical and Computational Theories of Vision - Modeling, Computing and Sampling* (Vancouver, BC).

Xie, S., Girshick, R., Dollar, P., Tu, Z., and He, K. (2017). "Aggregated residual transformations for deep neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition* (Honolulu).

Yang, B., Yan, J., Lei, Z., and Li, S. (2016). "CRAFT objects from images," in *IEEE Conference on Computer Vision and Pattern Recognition* (Las Vegas, NV).

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. (2017). "Understanding deep learning requires rethinking generalization," in *International Conference on Learning Representations* (Toulon).

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

frontiers in Genetics

ORIGINAL RESEARCH
published: 18 October 2018
doi: 10.3389/fgene.2018.00477

Deep Learning-Based Multi-Omics Data Integration Reveals Two Prognostic Subtypes in High-Risk Neuroblastoma

Li Zhang [1†], *Chenkai Lv* [1†], *Yaqiong Jin* [2,3†], *Ganqi Cheng* [1], *Yibao Fu* [1], *Dongsheng Yuan* [1], *Yiran Tao* [1], *Yongli Guo* [2,3*], *Xin Ni* [2,3,4*] *and Tieliu Shi* [1,4*]

[1] Center for Bioinformatics and Computational Biology, and the Institute of Biomedical Sciences, School of Life Sciences, East China Normal University, Shanghai, China, [2] Beijing Key Laboratory for Pediatric Diseases of Otolaryngology, Head and Neck Surgery, MOE Key Laboratory of Major Diseases in Children, Beijing Children's Hospital, National Center for Children's Health, Beijing Pediatric Research Institute, Capital Medical University, Beijing, China, [3] Biobank for Clinical Data and Samples in Pediatrics, Beijing Children's Hospital, National Center for Children's Health, Beijing Pediatric Research Institute, Capital Medical University, Beijing, China, [4] Department of Otolaryngology, Head and Neck Surgery, Beijing Children's Hospital, National Center for Children's Health, Capital Medical University, Beijing, China

OPEN ACCESS

Edited by:
Ping Gong,
Engineer Research and Development Center (ERDC), United States

Reviewed by:
Leihong Wu,
National Center for Toxicological Research (FDA), United States
Pawel Labaj,
Małopolska Centre of Biotechnology, Jagiellonian University, Poland
Ruili Huang,
National Center for Advancing Translational Sciences (NCATS), United States

***Correspondence:**
Tieliu Shi
tieliushi@yahoo.com
Xin Ni
nixin@bch.com.cn
Yongli Guo
guoyongli@bch.com.cn

† These authors have contributed equally to this work

Specialty section:
This article was submitted to Bioinformatics and Computational Biology, a section of the journal Frontiers in Genetics

Received: *28 July 2018*
Accepted: *26 September 2018*
Published: *18 October 2018*

Citation:
Zhang L, Lv C, Jin Y, Cheng G, Fu Y, Yuan D, Tao Y, Guo Y, Ni X and Shi T (2018) Deep Learning-Based Multi-Omics Data Integration Reveals Two Prognostic Subtypes in High-Risk Neuroblastoma. Front. Genet. 9:477. doi: 10.3389/fgene.2018.00477

High-risk neuroblastoma is a very aggressive disease, with excessive tumor growth and poor outcomes. A proper stratification of the high-risk patients by prognostic outcome is important for treatment. However, there is still a lack of survival stratification for the high-risk neuroblastoma. To fill the gap, we adopt a deep learning algorithm, Autoencoder, to integrate multi-omics data, and combine it with K-means clustering to identify two subtypes with significant survival differences. By comparing the Autoencoder with PCA, iCluster, and DGscore about the classification based on multi-omics data integration, Autoencoder-based classification outperforms the alternative approaches. Furthermore, we also validated the classification in two independent datasets by training machine-learning classification models, and confirmed its robustness. Functional analysis revealed that *MYCN* amplification was more frequently occurred in the ultra-high-risk subtype, in accordance with the overexpression of *MYC/MYCN* targets in this subtype. In summary, prognostic subtypes identified by deep learning-based multi-omics integration could not only improve our understanding of molecular mechanism, but also help the clinicians make decisions.

Keywords: deep learning, high-risk neuroblastoma, multi-omics data integration, *MYCN* amplification, machine learning

INTRODUCTION

Neuroblastoma is the most common extracranial solid tumor in childhood (mostly under the age of five) and accounts for approximately 15% of childhood cancer mortality (Ward et al., 2014). It can develop anywhere in the sympathetic nervous system (Maris et al., 2007). Sixty percent of the tumors occur within the abdomen, commonly in the adrenal medulla. The clinical hallmark of neuroblastoma is heterogeneity, with the outcomes of tumor progression varying widely. According to the Children's Oncology Group (COG) assignment, age at diagnosis, the stage of disease, *MYCN* amplification (Brodeur et al., 1984; Tomioka et al., 2008), the International

Neuroblastoma Pathology Classification and DNA ploidy are employed to stratify risk groups. Low-risk group has good outcome, whereas high-risk disease presents poor outcome even with the most intensive multi-modal therapies. Several recurrently mutated genes or loci which correlated with high-risk neuroblastoma have been identified, such as *ALK* (Mosse et al., 2008) mutations or amplifications, *PHOX2B* (Brodeur et al., 1984) mutation, chromosome 1p and 11q deletions, truncating or structural variants of *ATRX* gene (Cheung et al., 2012; Molenaar et al., 2012), genomic rearrangements of TERT (Peifer et al., 2015; Valentijn et al., 2015). These genetic events cover 92% of high-risk neuroblastoma (Peifer et al., 2015).

Driver genes/alterations, such as *MYCN*, 1p/11q deletion, *ALK*, *ATRX* and *TERT*, are characterized in high-risk neuroblastoma by previous studies, however, it is difficult to further stratify the high-risk neuroblastoma at molecular level. Previous studies have mostly intended to predict high-risk neuroblastoma survival using only genomic alterations (Stigliani et al., 2012) or dysregulated genes (Blanc et al., 2005; Chen et al., 2016; Wei et al., 2018), rarely by multi-omics integration. Therefore, the lack of prognostic stratification for high-risk neuroblastoma by multi-omics data integration motivated us to conduct this study.

With the production of omics data, such as The Cancer Genome Atlas (TCGA) and Therapeutically Applicable Research to Generate Effective Treatments (TARGET) projects, multi-omics integration is much needed in cancer researchers. Recently, Suo et al. (2018) have proposed a driver-gene score (DGscore) approach to predict the prognosis of the high-risk neuroblastoma by integrating the genome and transcriptome data. However, small sample size and no independent data for validation are the major limitations. Moreover, integrative clustering (iCluster) analysis (Shen et al., 2009; Cancer Genome Atlas Research, 2014) and PCA-based clustering analysis (Alexe et al., 2007; Nicolau et al., 2011) are widely applied to cancer subtyping. iCluster analysis could not only identify the molecular subtypes, but also associate the multi-omics data with each other. PCA is able to reduce the dimensionality of the multi-omics data, and integrates high dimensional multi-omics data into principal components. In addition, deep learning-based algorithm has been proposed to identify cancer subtypes. The recent study (Chaudhary et al., 2018) using deep learning-based multi-omics data integration robustly predicts survival in liver cancer. However, the multi-omics data integration approaches are rarely applied to neuroblastoma subtyping.

In this study, we used multi-omics-based unsupervised learning to stratify the high-risk neuroblastoma based on the new features re-encoded by Autoencoder algorithm, and compared the stratification with those identified by iCluster or PCA. The stratification of high-risk neuroblastoma by Autoencoder was also validated in two independent datasets, which may not only help the clinicians make rational and efficacious chemotherapeutic protocols, but also demonstrate that the deep learning-based algorithm is very efficient in multi-omics integration.

MATERIALS AND METHODS

Datasets and Study Design

We used multi-omics data from two projects in this study: Therapeutically Applicable Research to Generate Effective Treatments (TARGET) project (Pugh et al., 2013) and Sequencing Quality Control (SEQC) project (Zhang et al., 2015). The TARGET cohort is comprised of 407 high-risk neuroblastoma samples, including 217 samples with gene expression data and 380 samples with copy number alterations (CNA). Among these obtained samples, 190 has both gene expression and CNA data. The SEQC cohort has a total of 498 neuroblastoma samples, including 176 high-risk and 322 low- or intermediate-risk samples. The survival data were publicly available at the official website of TARGET project (https://ocg.cancer.gov/programs/target/data-matrix), and GEO database with accession number GSE49711 for SEQC cohort.

To integrate gene expression and CNA data, we first stacked these two datasets by the190 overlapping samples from TARGET cohort to form a new one. Then we selected the initial prognostic features (genes or CNAs) with Cox regression (log rank test, $P < 0.05$) for further analysis.

This new dataset with selected initial prognostic features was used in these following parts of our work: generating new features from a classic artificial neural network: Autoencoder (with which 100 new features were generated and Cox regression was applied again here to ensure that they were significantly prognostic), obtaining labels for different survival-risk groups through K-means clustering from transformed new features by Autoencoder, and training classifiers with models such as SVM, Naïve Bayes, and logistic regression according to the class labels.

Also, to demonstrate the robustness of the classification at predicting prognosis, these supervised classification methods mentioned above, along with XGBoost, were trained on gene expression data and CNA data respectively by different machine learning methods.

There were two datasets used for demonstrating the robustness of the classification for predicting prognosis, one was the remaining 190 samples which had CNA data only (the internal validation set), and the other was 176 high-risk samples with gene expression data in SEQC project (the external validation set). The class labels for samples from TARGET internal validation set and SEQC external validation set were predicted by CNA-based XGBoost and gene expression-based SVM models, respectively.

Gene Expression Data From Target and SEQC Projects

The gene expression data from TARGET project were profiled by Affymetrix Exon ST platform, and normalized by Robust Multi-array Average (RMA) procedure, which could be downloaded from the website (https://ocg.cancer.gov/programs/target/data-matrix).

As reported by Zhang et al. (2015), a total of 498 neuroblastoma samples were selected for RNA sequencing, of which, 176 high-risk neuroblastoma cases were selected for external validation. The RNA sequencing reads of 176 high-risk neuroblastoma samples were mapped to human reference genome GRCh37/hg19 with GENCODE gene annotation v19 by hisat2. The gene expression were then quantified by StringTie (Pertea et al., 2015) with default options and combined in R programming software with *ballgown* package.

Data Integration and Re-coding by Autoencoder

Autoencoder is a dimensionality reduction method based on artificial neural network, which consists of input, hidden, and output layers. The data integration analysis by Autoencoder was implemented in R programming software with package *ANN2*. To better capture properties that reflect the variation of patients' prognosis, a classic autoencoder with 3 hidden layers was applied (500, 100, and 500 nodes, respectively), of which the 100-node bottleneck layer was used to represent new features for further analysis. We then selected 35 survival-associated features (log-rank test, *P*-value < 0.05) from the 100 new features.

For a given layer, a specific activation function was assigned, and the output x' was given by a composite function of x, which was composed of all these activation functions from each layer, and could be expressed as:

$$\begin{aligned} \gamma &= f_i(x) = \tanh\left(W_i.x + b_i\right) \\ x' &= F_{1\to k}(x) = f_1 \ldots f_{k-1} f_k(x), \end{aligned}$$

where k represents to the number of layers.

We measured the error with function *the Pseudo-Huber loss function*, which ensures that derivatives are continuous for all degrees, that is:

$$L\left(x, x'\right) = \sum_{k=1}^{n} \left[\delta^2 \sqrt{1 + (\frac{x_k - x'_k}{\delta})2} - 1\right]$$

where n stands for the dimension of the input data. As can be seen in **Supplementary Figure S1**, the output data from reconstructed layer was compared with the raw input data with Pseudo-Huber loss function.

The Autoencoder was trained using the gradient descent algorithm with 10 epochs, a batch size of 32, and a learning rate of 1e-6. The parameters of L1 and L2 regularization were set to 0.0001 and 0.001.

Gene Expression Data Normalization

For the RMA-based gene expression data by microarray platform, we transformed the expression value as Z-score for each gene. For the gene expression data of RNA-seq, we firstly calculated the fractions of the genes that had a FPKM value over the threshold we set, which can be seen in the **Supplementary Table S5** (We selected several possible thresholds, e.g., 0.01, 0.05, 0.1, 0.5, 1). We then applied an alternative method, instead of adding an arbitrary value, we calculated the minimal value for each sample which is not zero, and then set all values below maximum of these minimums (which is 3.7e-05) to the minimum of these minimums (which is 1.60e-07) in all samples, and then transformed by logarithm with base-2. Like the gene expression data by microarray platform, the gene expression values were also transformed to Z-scores in similar manner.

CNA Data Annotation

The segmented copy number regions with segment means were available at the TARGET website (https://ocg.cancer.gov/programs/target/data-matrix). We merged the segmented CNAs from the 380 samples, and annotated the genes in the CNAs by GISTIC2.0 (Mermel et al., 2011), which is implemented in GenePattern (Reich et al., 2006), a webserver publicly available for researchers (https://software.broadinstitute.org/cancer/software/genepattern/). The rows and columns of the CNA matrix represent the genes and samples, respectively. Each element of the CNA matrix was normalized as $\log_2$ (segmented copy number)−1.

Feature and Model Selection

To integrate the multi-omics data, we applied three methods: autoencoder-based deep learning, iCluster and PCA, and then we compared the labels identified by these three approaches. Unlike iCluster, autoencoder-based deep learning and PCA were not clustering algorithms, thus the other two were followed by k-means clustering. Taking together, these three methods were able to integrate multi-omics data and were evaluated by the association between classification labels and patients' prognosis.

Machine learning classifiers such as SVM, Naïve Bayes, and logistic regression are supervised learning algorithms. The classification labels used for these machine learning classifiers were only determined by autoencoder-based deep learning followed by K-means clustering, not by the other methods. After obtaining the labels from K-means clustering, we need to examine the robustness of this sample stratification. We then built two supervised models based on the gene expression and CNA data, respectively, and predicted the classes for samples from both internal and external validation sets. The machine-learning classification models were then used to test its robustness in validation sets.

Features for the models, including Naïve Bayes, logistic regression and SVM, were selected by a backward elimination manner. For each gene or CNA, the importance was evaluated by the ANOVA *F*-value. 10-fold cross-validation with 10-time repeat was conducted to evaluate the predictive ability of the selected features (genes or CNAs). The feature combinations with highest average predictive accuracy were selected. The features for XGBoost were selected by its internal algorithm. Given the features, receiver operating characteristic (ROC) curve was plotted for each model, and the one with highest area under the curve (AUC) was selected as the prediction model.

Statistical Analysis

The statistical analyzing methods such as Cox proportional hazards (Cox-PH) analysis, principal component analysis, K-means clustering, integrative clustering and student-*t* test were implemented in R programming software with version 3.5.0. In addition, we determined the optimal number of clusters on three metrics: C index for the prognostic differences, Silhouette index and Calinski–Harabasz criterion (**Supplementary Table S4**). The overrepresentation enrichment analysis (OEA) was implemented in WebGestalt (Wang et al., 2017) (http://www.webgestalt.org/option.php) with a functional database named Hallmark50 (Liberzon et al., 2015).

RESULTS

Data Collection and Pre-processing For Integrative Analysis

We collected 407 high-risk neuroblastoma samples from TARGET project (Ma et al., 2018), including 217 samples with gene expression data and 380 samples with copy number alterations (CNA). The neuroblastoma patients were treated according to Children's Oncology Group (COG) risk-group assignment. Among the obtained tumor samples, 190 had both gene expression and CNA data, which were used as training data in this study. Multi-omics data in the training data were integrated to discover a prognostic stratification of the high-risk neuroblastoma. The remaining 190 samples with only CNA data were used as an internal validation data to test the robustness of classification. In addition, we also collected RNA sequencing data of 176 high-risk neuroblastoma samples from SEQC project, which was used as an external validation data to further test the robustness.

As illustrated in **Figure 1**, prior to multi-omics integration, prognosis-associated genes were selected from both gene expression and CNA data of the 190 NB samples based on the univariate Cox proportional hazards (Cox-PH) regression analysis. Finally, 2,218 aberrantly expressed genes and 497 copy number altered genes were associated with the prognosis of high-risk neuroblastoma [*P*-value < 0.05 for event-free survival (EFS) or overall survival (OS)], which were used for integrative analysis later on.

Identification of Prognostic Subtypes in High-Risk Neuroblastoma

To identify the prognostic subtypes in high-risk neuroblastoma, we stacked the two matrices of gene expression and CNA by the 190 overlapping samples in TARGET project, and transformed the initial prognostic features into 100 new features according to Autoencoder, a five-layer neural network with three hidden layers (500, 100, and 500 nodes). The two-omics data were integrated and represented by the 100 new features obtained from the bottleneck layer of the autoencoder. We then conducted a univariate Cox-PH regression on each of the 100 new features and identified 35 features significantly ($P < 0.05$) associated with EFS or OS. Subsequently, K-means clustering analysis was performed on the 35 new features with clustering number ranging from 2 to 6 (**Figure 1**). We determined the optimal number of clusters based on three metrics: C index for the prognostic differences, Silhouette index and Calinski–Harabasz criterion, which consistently supported our choice of 2 as

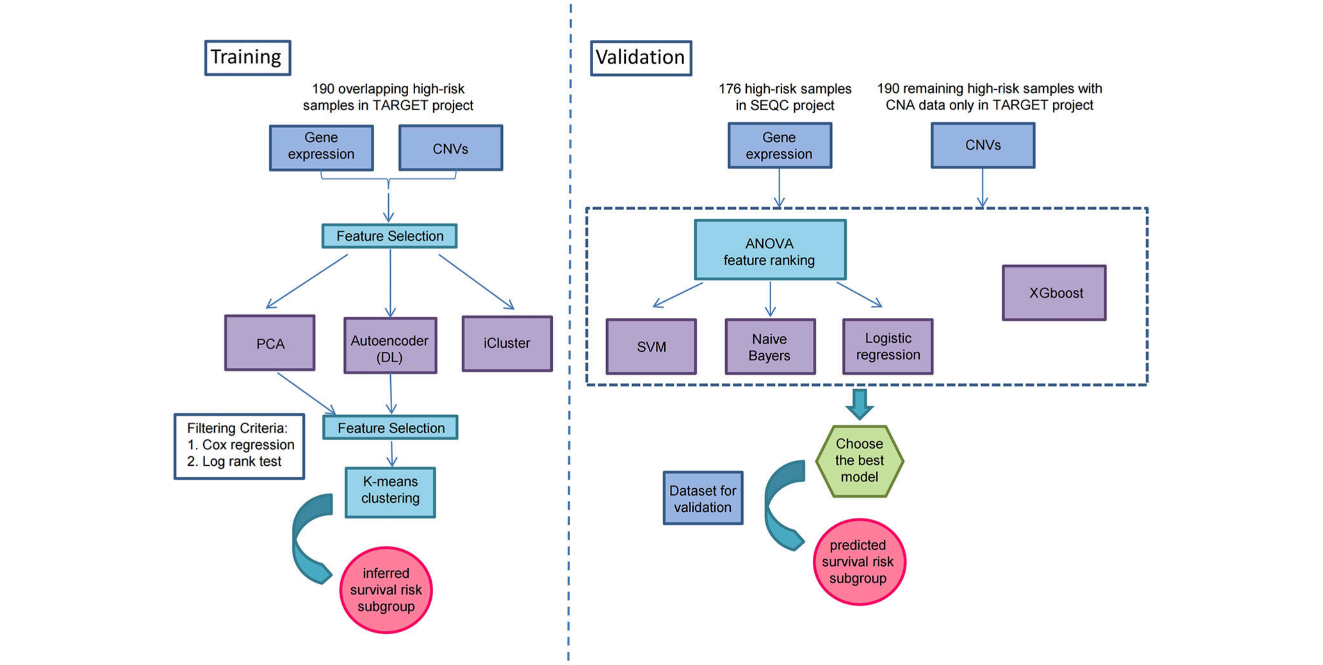


FIGURE 1 | Overview workflow for the identification of prognostic subtypes by Autoencoder-based multi-omics data integration in high-risk neuroblastoma.

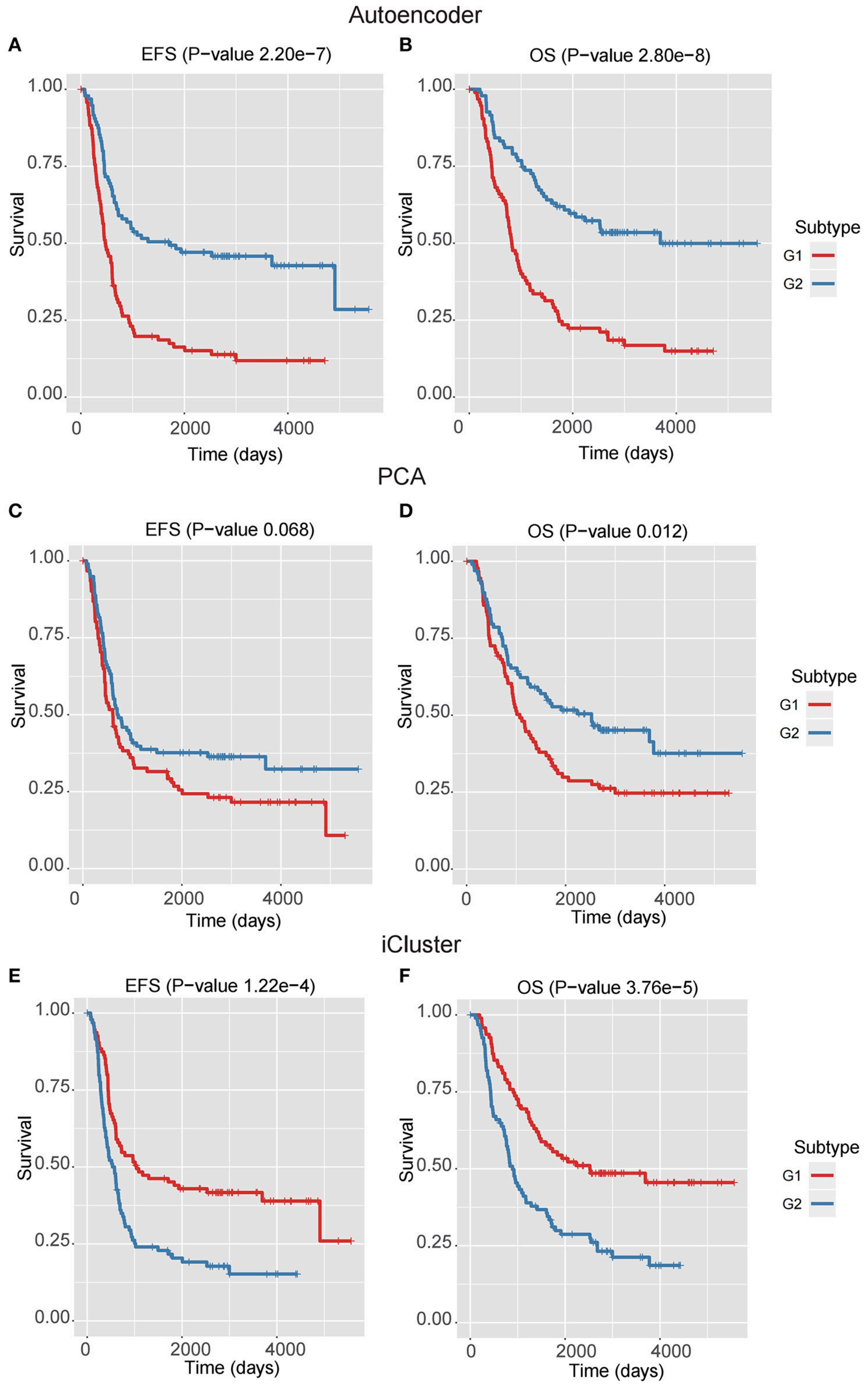


FIGURE 2 | The Kaplan–Meier curves for EFS or OS of two identified subtypes by three multi-omics integration algorithms, Autoencoder **(A,B)**, PCA **(C,D)**, and iCluster **(E,F)**.

the number of clusters (**Supplementary Table S4**). Finally, we clustered the samples into two subtypes, which were defined as G1 and G2.

We next assessed the prognostic difference between these two subgroups by univariate Cox-PH regression, and observed that the G1 exhibited worse prognosis than G2 (P-value < 0.0001 for both EFS and OS, **Figures 2A,B**), indicating that G1 was an ultra-high-risk subtype. Moreover, the concordance index (C-index), which measures the fraction of all pairs of cases whose predicted survival times are ordered correctly, was also calculated. Expectedly, our classification also generated high C-index (0.74 ± 0.08 for EFS and 0.71 ± 0.08 for OS). The result indicated that our classification revealed two prognostic subtypes in high-risk neuroblastoma.

Autoencoder-Based Multi-Omics Integration Outperforms Alternative Approaches

In addition to Autoencoder-based multi-omics integration, principal component analysis (PCA) and integrative clustering analysis (iCluster) were also incorporated to evaluate the performance of multi-omics integration approaches (**Supplementary Table S1**). Similar to the 100 new features by Autoencoder, PCA transformed the inital features into 100 principal components, and Cox-PH was applied to select prognostic principal components. As a result, 14 principal components were remained. Unlike PCA and Autoencoder, the iCluster analysis did not have to transform the initial prognostic features into new features, but placed cases into groups based on both gene expression patterns and copy number status.

In the training data, we found that the classification by Autoencoder had better performance than the other two approaches (**Figures 2C–F**), among which iCluster achieved high C-index and significant log-rank P-value, but it was still less significant as compared with the model using Autoencoder, and the PCA-based classification showed poor performance, especially failing to give significant log-rank P value for EFS ($P = 0.068$). In addition, as compared with the DGscore method (P-value $= 0.006$), Autoencoder-based classification also achieved higher statistical significance ($P = 5.66\text{e-}6$ for EFS and $P = 1.28\text{e-}5$ for OS). The result indicated that Autoencoder-based multi-omics integration outperformed these alternative approaches.

TABLE 1 | Performance of four classifiers using the training dataset.

Feature selection + classifier	Feature type	Feature number	AUC	Average accuracy	Average AUC
ANOVA + SVM	GE	56	0.9962	0.7553	0.8446
	CNA	30	0.6586	0.5937	0.5159
ANOVA + naïve bayes	GE	46	0.9299	0.6755	0.8291
	CNA	24	0.6019	0.5234	0.5506
ANOVA + logistic regression	GE	44	0.9703	0.7059	0.6053
	CNA	15	0.6782	0.6135	0.5699
Xgboost	GE	64	0.9602	0.7338	0.8025
	CNA	30	0.954	0.6559	0.6317

GE, gene expression; CAN, copy number alteration; ANOVA, analysis of variance; SVM, support vector machine; AUC, area under the curve; Average accuracy, average of the accuracies from 10-fold cross-validation. Average AUC, average of the AUC values from 10-fold cross-validation.

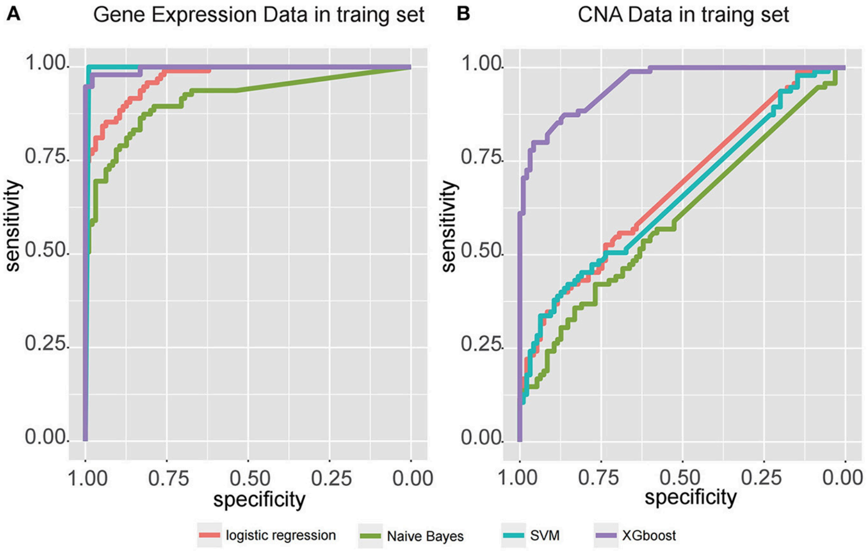


FIGURE 3 | Receiver operating characteristic (ROC) curve for four classifiers, including logistic regression, Naïve Bayes, SVM, and XGBoost, that predict the subtypes of samples from two independent datasets, **(A)** gene expression data from SEQC external validation cohort, and **(B)** CNA data from TARGET internal validation cohort.

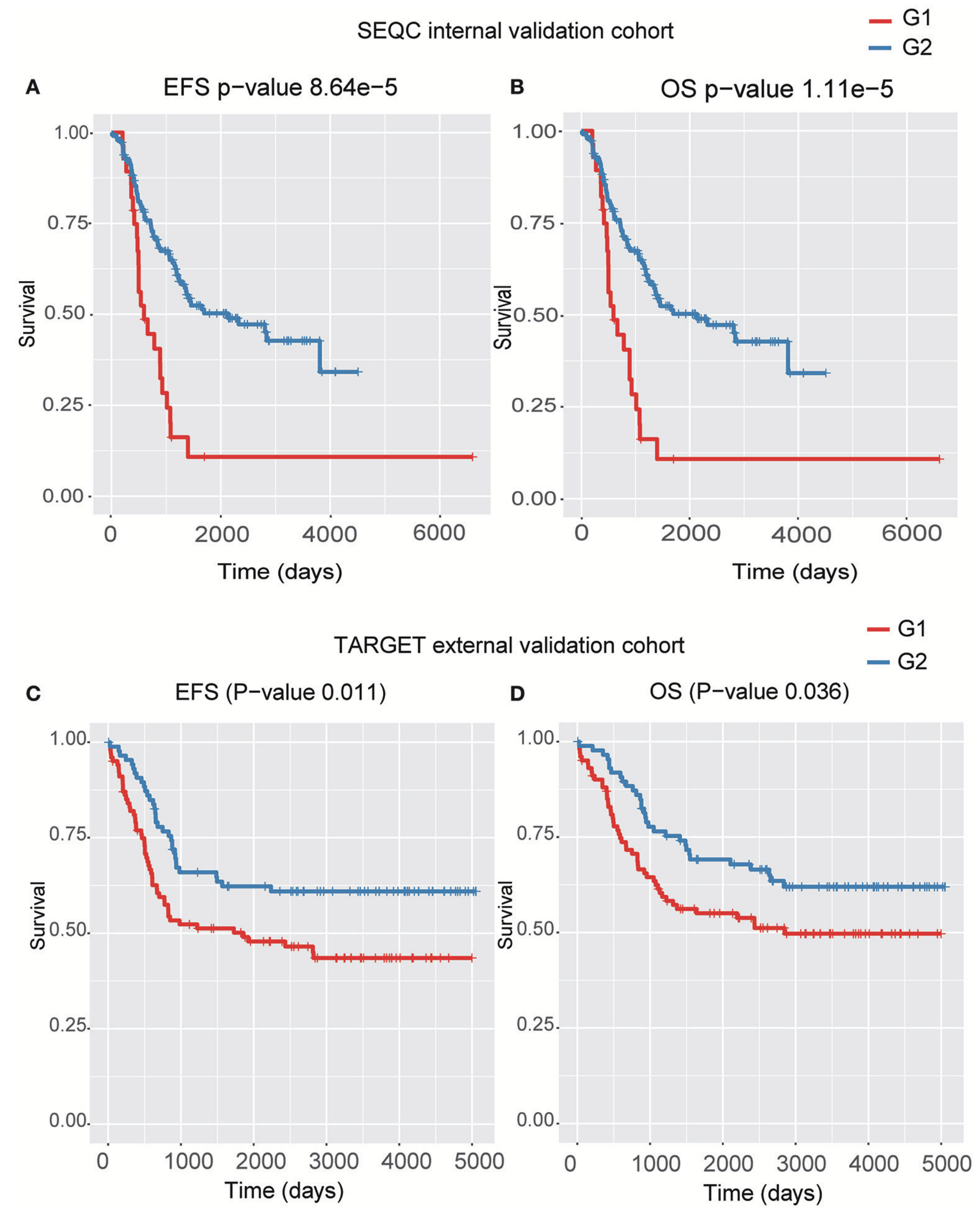


FIGURE 4 | The Kaplan–Meier curves for EFS or OS of two predicted subtypes for the high-risk tumors from SEQC external validation cohort **(A,B)** and TARGET internal validation **(C,D)** cohort.

Prognostic Subtypes Are Validated in Two Validation Datasets

To demonstrate the robustness of the classification at predicting prognosis, we built two supervised classification models based on gene expression and CNA data separately to predict the classification labels for samples from both internal and external validation datasets, respectively.

After obtaining the labels from K-means clustering, we first built two supervised models based on the gene expression and CNA data, respectively. Each omics data was normalized as Z-score to avoid platform differences. For the internal validation, we used the remaining 190 samples with only CNA data from the TARGET project which didn't overlap with the samples with gene expression data. Meanwhile, the 176 SEQC high-risk neuroblastoma samples with gene expression data was used as external validation.

Four models, including SVM, naïve Bayes, logistic regression, and XGBoost, were built to select the best model for classification prediction. Based on ten-fold cross-validation in the training dataset, SVM exhibited high capability of predicting classification

TABLE 2 | Hallmark gene sets identified by OEA (FDR < 0.05).

Status	Gene set	Description	*P*-value	FDR
Up	HALLMARK_MYC_TARGETS_V2	MYC targets, variant 2	9.81E-07	4.9E-05
Down	HALLMARK_INTERFERON_ALPHA_RESPONSE	Interferon-alpha response	5.14E-03	5.76E-01

labels for 176 samples from the external validation set using gene expression data (See features in **Supplementary Table S2**), while XGBoost achieved higher performance on the CNA data than other models (**Figure 3**, **Table 1**, and see features in **Supplementary Table S3**). For the gene expression data from SEQC project, we achieved good C-indices (0.69 ± 0.08 for EFS and 0.74 ± 0.08 for OS) and log-rank *P* values (< 0.0001) between the two subtypes (**Figures 4A,B**). For the CNV data from TARGET internal validation cohort, the classification had C-indices over 0.64 and low log-rank *P* values ($P < 0.05$, **Figures 4C,D**). The validation of the classification in both internal and external datasets further demonstrated that the two subtypes indeed had different outcomes.

Functional Analysis of the Prognostic Subtypes in High-Risk NB

We used *t*-test for differential gene expression between the two subtypes of both training and validation datasets. At the FDR < 0.05 for both datasets, we obtained 302 upregulated and 851 downregulated genes in the subtype G1. Overrepresentation enrichment analysis (OEA) was then performed on the two gene sets (**Table 2**). We identified MYC target genes, including *FARSA*, *SLC29A2*, *PLK1*, *WDR74*, *RRP9*, and *IMP4*, were upregulated in G1 subtype (FDR < 0.05). Interestingly, *MYCN* amplification (MNA) was observed to present higher frequency in G1 than G2 ($P = 0.054$, 35 vs. 26% in training data, and $P < 0.005$, 77 vs. 44% in the validation data), indicating that our classification was associated with MNA to a certain extent, but some samples without MNA also had poor survival. However, we did not identify significant down-regulated pathways in G1 subtype. Alternatively, interferon-alpha response pathway was down-regulated in G1 (*P*-value < 0.05), which is a common defect in human cancers (Critchley-Thorne et al., 2009). In detail, the genes in interferon-alpha response pathway, such as CMTR1, NUB1, and STAT2, were consistently down-regulated in G1 subtype. The result indicated that interferon-alpha may be a potential immunotherapy strategy for the ultra-high risk neuroblastoma.

DISCUSSION

Recently, with the development of high-throughput technologies, such as DNA microarray, next generation sequencing, and mass spectrum-based proteomics, huge amounts of omics data are produced and made available publicly. However, high production of multi-omics data also raises requirements to comprehensively analyze different levels of omics data.

In the present study, we have adopted a deep learning-based algorithm, Autoencoder, to integrate copy number alterations and gene expression data to identify two prognostic subtypes, defined as G1 and G2, in high-risk neuroblastoma. The subtype G1 exhibits worse prognosis than G2 in both EFS and OS (*P*-value < 0.0001). The Autoencoder-based classification also generates high C-index (0.74 ± 0.08 for EFS and 0.71 ± 0.08 for OS). The performance comparison of Autoencoder with PCA and iCluster demonstrates that our Autoencoder-based classification is superior to the two alternative approaches. Moreover, the result of Autoencoder-based classification is also more significant than DGscore method. To demonstrate the robustness of the classification, we build two supervised classifiers for the independent CNA and gene expression datasets, respectively. For both of the datasets, we achieve good C-indices and significant log-rank *P*-values ($P < 0.05$). We thus conclude that Autoencoder-based classification outperforms other approaches, and we speculate that the unique advantage of the Autoencoder, which can capture the core features relevant to the prognosis, have contributed to this.

High-risk neuroblastoma is an aggressive disease. To our knowledge, the present study is the first to apply deep learning approach to distinguish ultra-high-risk subgroup from the high-risk neuroblastoma, with validation in independent datasets. The integrative classification of the high-risk neuroblastoma may help clinicians develop personalized treatment programs, and better predict patients' prognosis.

CONCLUSION

Prognostic subtypes identified by deep learning-based multi-omics integration could not only improve our understanding of molecular mechanism, but also help the clinicians make decisions.

AUTHOR CONTRIBUTIONS

In this study, TS designed the study. LZ and CL adapted algorithms and software for building model and data analysis. LZ, CL, and YJ interpreted data in context of NB biology. LZ, YJ, CL, and GC drafted the manuscript. YF, DY, and YT collected the data. TS, XN, and YG revised and finalized the manuscript. All authors read and approved the final manuscript.

FUNDING

This work was supported by the China Human Proteome Project (Grant No.2014DFB30010, 2014DFB30030), National Key Research and Development Program of China (2016YFC0902100), National Natural Science Foundation of China (31671377, 81472369), Shanghai 111 Project (B14019) and Clinical Application Research Funds of Capital Beijing (Z171100001017051). This work is also supported by National Natural Science Foundation of China (81502144 and 81472369).

ACKNOWLEDGMENTS

We thank the supercomputer center of East China Normal University for their support.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fgene.2018.00477/full#supplementary-material

Supplementary Figure S1 | The schematic diagram for Autoencoder algorithm.

Supplementary Table S1 | Prognostic subtypes identified by three algorithms.

Supplementary Table S2 | Gene features used by SVM.

Supplementary Table S3 | CNA genes selected by XGBoost.

Supplementary Table S4 | Scores of different cluster numbers.

Supplementary Table S5 | Percentages of genes among different thresholds based on expression.

REFERENCES

Alexe, G., Dalgin, G. S., Ganesan, S., Delisi, C., and Bhanot, G. (2007). Analysis of breast cancer progression using principal component analysis and clustering. *J Biosci.* 32, 1027–1039. doi: 10.1007/s12038-007-0102-4

Blanc, E., Roux, G. L., Benard, J., and Raguenez, G. (2005). Low expression of Wnt-5a gene is associated with high-risk neuroblastoma. *Oncogene* 24, 1277–1283. doi: 10.1038/sj.onc.1208255

Brodeur, G. M., Seeger, R. C., Schwab, M., Varmus, H. E., and Bishop, J. M. (1984). Amplification of N-myc in untreated human neuroblastomas correlates with advanced disease stage. *Science* 224, 1121–1124.

Cancer Genome Atlas Research, N. (2014). Comprehensive molecular profiling of lung adenocarcinoma. *Nature* 511, 543–550. doi: 10.1038/nature 13385

Chaudhary, K., Poirion, O. B., Lu, L., and Garmire, L. X. (2018). Deep learning-based multi-omics integration robustly predicts survival in liver cancer. *Clin. Cancer Res.* 24, 1248–1259. doi: 10.1158/1078-0432.CCR-17-0853

Chen, G., Yang, J., Chen, J., Song, Y., Cao, R., Shi, T., et al. (2016). Identifying and annotating human bifunctional RNAs reveals their versatile functions. *Sci. China. Life Sci.* 59, 981–992. doi: 10.1007/s11427-016-0054-1

Cheung, N. K., Zhang, J., Lu, C., Parker, M., Bahrami, A., Tickoo, S. K., et al. (2012). Association of age at diagnosis and genetic mutations in patients with neuroblastoma. *JAMA* 307, 1062–1071. doi: 10.1001/jama.2012.228

Critchley-Thorne, R. J., Simons, D. L., Yan, N., Miyahira, A. K., Dirbas, F. M., Johnson, D. L., et al. (2009). Impaired interferon signaling is a common immune defect in human cancer. *Proc. Nat. Acad. Sci. U.S.A.* 106, 9010–9015. doi: 10.1073/pnas.0901329106

Liberzon, A., Birger, C., Thorvaldsdóttir, H., Ghandi, M., Mesirov, J. P., Tamayo, P. (2015). The molecular signatures database (MSigDB) hallmark gene set collection. *Cell Syst.* 1, 417–425. doi: 10.1016/j.cels.2015.12.004

Ma, X., Liu, Y., Liu, Y., Alexandrov, L. B., Edmonson, M. N., Gawad, C., et al. (2018). Pan-cancer genome and transcriptome analyses of 1,699 paediatric leukaemias and solid tumours. *Nature* 555, 371–376. doi: 10.1038/nature25795

Maris, J. M., Hogarty, M. D., Bagatell, R., and Cohn, S. L. (2007). Neuroblastoma. *Lancet* 369, 2106–2120. doi:10.1016/S0140-6736(07)60983-0

Mermel, C. H., Schumacher, S. E., Hill, B., Meyerson, M. L., Beroukhim, R., Getz, G. (2011). GISTIC2.0 facilitates sensitive and confident localization of the targets of focal somatic copy-number alteration in human cancers. *Genome Biol.* 12:R41. doi: 10.1186/gb-2011-12-4-r41

Molenaar, J. J., Koster, J., Zwijnenburg, D. A., van Sluis, P., Valentijn, L. J., van der Ploeg, I., et al. (2012). Sequencing of neuroblastoma identifies chromothripsis and defects in neuritogenesis genes. *Nature* 483, 589–593. doi: 10.1038/nature10910

Mosse, Y. P., Laudenslager, M., Longo, L., Cole, K. A., Wood, A., Attiyeh, E. F., et al. (2008). Identification of ALK as a major familial neuroblastoma predisposition gene. *Nature* 455, 930–935. doi: 10.1038/nature07261

Nicolau, M., Levine, A. J., and Carlsson, G. (2011). Topology based data analysis identifies a subgroup of breast cancers with a unique mutational profile and excellent survival. *Proc. Nat. Acad. Sci. U S A.* 108, 7265–7270. doi: 10.1073/pnas.1102826108

Peifer, M., Hertwig, F., Roels, F., Dreidax, D., Gartlgruber, M., Menon, R., et al. (2015). Telomerase activation by genomic rearrangements in high-risk neuroblastoma. *Nature* 526, 700–704. doi: 10.1038/nature14980

Pertea, M., Pertea, G. M., Antonescu, C. M., Chang, T. C., Mendell, J. T., Salzberg, S. L. (2015). StringTie enables improved reconstruction of a transcriptome from RNA-seq reads. *Nat. Biotechnol.* 33, 290–295. doi: 10.1038/nbt.3122

Pugh, T. J., Morozova, O., Attiyeh, E. F., Asgharzadeh, S., Wei, J. S., Auclair, D., et al. (2013). The genetic landscape of high-risk neuroblastoma. *Nat. Genet.* 45, 279–284. doi: 10.1038/ng.2529

Reich, M., Liefeld, T., Gould, J., Lerner, J., Tamayo, P., Mesirov, J. P. (2006). GenePattern 2.0. *Nat. Genet.* 38, 500–501. doi: 10.1038/ng0506-500

Shen, R., Olshen, A. B., and Ladanyi, M. (2009). Integrative clustering of multiple genomic data types using a joint latent variable model with application to breast and lung cancer subtype analysis. *Bioinformatics* 25, 2906–2912. doi: 10.1093/bioinformatics/btp543

Stigliani, S., Coco, S., Moretti, S., Oberthuer, A., Fischer, M., Theissen, J., et al. (2012). High genomic instability predicts survival in metastatic high-risk neuroblastoma. *Neoplasia* 14, 823–832. doi: 10.1593/neo.21114

Suo, C., Deng, W., Vu, T. N., Li, M., Shi, L., Pawitan, Y. (2018). Accumulation of potential driver genes with genomic alterations predicts survival of high-risk neuroblastoma patients. *Biol. Direct* 13:14. doi: 10.1186/s13062-018-0218-5

Tomioka, N., Oba, S., Ohira, M., Misra, A., Fridlyand, J., Ishii, S., et al. (2008). Novel risk stratification of patients with neuroblastoma by genomic signature, which is independent of molecular signature. *Oncogene* 27, 441–449. doi: 10.1038/sj.onc.1210661

Valentijn, L. J., Koster, J., Zwijnenburg, D. A., Hasselt N. E., van Sluis P., Volckmann, R., et al. (2015). TERT rearrangements are frequent in neuroblastoma and identify aggressive tumors. *Nat. Gene.* 47, 1411–1414. doi: 10.1038/ng.3438

Wang, J., Vasaikar, S., Shi, Z., Greer, M., and Zhang, B. (2017). WebGestalt 2017: a more comprehensive, powerful, flexible and interactive gene set enrichment analysis toolkit. *Nucleic Acids Res.* 45, W130–W137. doi: 10.1093/nar/gkx356

Ward, E., DeSantis, C., Robbins, A., Kohler, B., and Jemal (2014). A. Childhood and adolescent cancer statistics, 2014. *CA Cancer J. Clin.* 64, 83–103. doi: 10.3322/caac.21219

Wei, J. S., Kuznetsov, I. B., Zhang, S., Song, Y. K., Asgharzadeh, S., Sindiri, S., et al. (2018). Clinically relevant cytotoxic immune cell signatures and clonal expansion of T cell receptors in high-risk MYCN-not-amplified human neuroblastoma. *Clin. Cancer Res.* doi: 10.1158/1078-0432.CCR-18-0599. [Epub ahead of print].

Zhang, W., Yu, Y., Hertwig, F., Thierry-Mieg, J., Zhang, W., Thierry-Mieg, D., et al. (2015). Comparison of RNA-seq and microarray-based models for clinical endpoint prediction. *Genome Biol.* 16:133. doi: 10.1186/s13059-015-0694-1

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

ORIGINAL RESEARCH
published: 27 November 2018
doi: 10.3389/fgene.2018.00585

Prediction of Drug-Likeness Using Deep Autoencoder Neural Networks

Qiwan Hu[1], Mudong Feng[2], Luhua Lai[1,2] and Jianfeng Pei[1]*

[1] Center for Quantitative Biology, Academy for Advanced Interdisciplinary Studies, Peking University, Beijing, China,
[2] BNLMS, State Key Laboratory for Structural Chemistry of Unstable and Stable Species, Peking-Tsinghua Center for Life Sciences at College of Chemistry and Molecular Engineering, Peking University, Beijing, China

Due to diverse reasons, most drug candidates cannot eventually become marketed drugs. Developing reliable computational methods for prediction of drug-likeness of candidate compounds is of vital importance to improve the success rate of drug discovery and development. In this study, we used a fully connected neural networks (FNN) to construct drug-likeness classification models with deep autoencoder to initialize model parameters. We collected datasets of drugs (represented by ZINC World Drug), bioactive molecules (represented by MDDR and WDI), and common molecules (represented by ZINC All Purchasable and ACD). Compounds were encoded with MOLD2 two-dimensional structure descriptors. The classification accuracies of drug-like/non-drug-like model are 91.04% on WDI/ACD databases, and 91.20% on MDDR/ZINC, respectively. The performance of the models outperforms previously reported models. In addition, we develop a drug/non-drug-like model (ZINC World Drug vs. ZINC All Purchasable), which distinguishes drugs and common compounds, with a classification accuracy of 96.99%. Our work shows that by using high-latitude molecular descriptors, we can apply deep learning technology to establish state-of-the-art drug-likeness prediction models.

Keywords: drug-likeness, ZINC, MDDR, deep learning, auto-encoder

OPEN ACCESS

Edited by:
Ping Gong,
Engineer Research and Development Center (ERDC), United States

Reviewed by:
Yun Tang,
East China University of Science and Technology, China
Shengyong Yang,
Sichuan University, China

****Correspondence:***
Jianfeng Pei
jfpei@pku.edu.cn

Specialty section:
This article was submitted to Toxicogenomics, a section of the journal Frontiers in Genetics

Received: *31 August 2018*
Accepted: *09 November 2018*
Published: *27 November 2018*

Citation:
Hu Q, Feng M, Lai L and Pei J (2018) Prediction of Drug-Likeness Using Deep Autoencoder Neural Networks. Front. Genet. 9:585. doi: 10.3389/fgene.2018.00585

INTRODUCTION

Over the past several decades, various novel and effective techniques, such as high-throughput screening(HTS), fragment-based drug discovery (FBDD), single-cell analysis, have been developed and led to remarkable progresses in the field of drug discovery. However, it is noted that the amount of new chemical entities (NCEs) approved by FDA did not grow as rapidly as expected (Darrow and Kesselheim, 2014). According to statistics, the success rate of candidate compounds found in preclinical detection is about 40%, while the rate of candidate compounds entering the market is only 10% (Lipper, 1999).

About 40% of the candidate compounds not being marketed is due to their poor biopharmaceutical properties, also commonly referred to as drug-likeness, which includes poor chemical stability, poor solubility, poor permeability and poor metabolic (Venkatesh and Lipper, 2000). Drug-likeness, derived from structures and properties of existing drugs and drug candidates,

Abbreviations: 5-CV, 5-fold cross-validation; ACC, accuracy; AE, autoencoder; AUC, areas under the receiver operating characteristic curve; DL, deep learning; DNN, deep neural network; FNN, fully connected neural network; MCC, Matthews correlation coefficient; SAE, stacked autoencoder; SE, sensitivity; SP, specificity; SVM, support vector machine.

has been widely used to filter out undesirable compounds in early phases of drug discovery. The initial concept of drug-like rules is proposed by Lipinsky, known as the rule-of-five which contains four simple physicochemical parameter definitions (MWT $\leq$ 500, log P $\leq$ 5, H-bond donors $\leq$ 5, H-bond acceptors $\leq$ 10) (Lipinski, 2004). Using these definitions may predict whether a compound can become an oral drug candidate. In 2012, Hopkins et al. propose the quantitative estimate of drug-likeness (QED) measure, which was a weighted desirability function based on the statistical distribution of eight selected molecular properties for a set of 771 orally absorbed small molecule drugs and applied to molecular target druggability assessment (Bickerton et al., 2012). Due to the ambiguous definition of molecular properties between the drugs and non-drug and the prediction is not satisfactory with few descriptors, later works tried to combine more comprehensive descriptors and a large amount of compound data to develop drug-likeness prediction models with high accuracies from a quantitative perspective.

A drug-likeness prediction model introduced by Wagener et al., involved molecular descriptors related to numbers of different atom types and decision trees for discriminating between potential drugs and nondrugs. The model was trained using 10,000 compounds from the ACD and the WDI, and its prediction ACC on an independent validation data set of 177,747 compounds was 82.6% (Wagener and van Geerestein, 2000). In 2003, Byvatov and co-workers used various different descriptor sets and descriptor combinations to characterize compound and applied SVM and artificial neural network (ANN) systems to solve the drug/nondrug classification problem. Both methods reached 80% correct predictions and their results indicated SVM seemed to be more robust (Byvatov et al., 2003). A later model reported by Muller was also based on SVM with a careful model selection procedure for improving the prediction results of Byvatov et al. (2003) (Müller et al., 2005). In 2008, Li et al implemented ECFP_4 (Extended Connectivity Fingerprints) for characterizing the molecules and used a probability SVM model to classify drug-like and non-drug-like molecules. The model significantly improved the prediction ACC when compared to previous work on the same data sets, and it is surprising that when using a larger data set of 341,601 compounds the classifier increased the ACC to 92.73% (Li et al., 2007). Schneider et al. applied decision trees to perform a gradual *in silico* screening for drug-like compounds based on SMARTS strings and the molecular weight, XlogP, and the molar refractivity as descriptor space for compounds (Schneider et al., 2008). In 2012, Tian et al implemented 21 physicochemical properties and the LCFP_6 fingerprint encoding molecules and used the naive Bayesian classification (NBC) and recursive partitioning (RP) to construct drug-like/non-drug-like classifier, which achieved 90.9% ACC (Tian et al., 2012). These studies showed that machine learning techniques are highly potential for the drug-likeness prediction problem combined with big data sets.

Deep learning is a new wave of machine learning based on artificial neural networks (ANN) (Bengio, 2009; Vincent et al., 2010). Since 2006, DL has been showing superior performances in many fields, such as computer vision (Hinton et al., 2006; Coates et al., 2011; Krizhevsky et al., 2012; He et al., 2016), natural language processing (Dahl et al., 2012; Socher et al., 2012; Graves et al., 2013; Mikolov et al., 2013; Bahdanau et al., 2016), bioinformatics and chemoinformatics (Di Lena et al., 2012; Lyons et al., 2014; Heffernan et al., 2015; Chen et al., 2016; Zeng et al., 2016). Compared to traditional machine learning methods, DL with multiple levels of layers can automatically transform raw data into a suitable internal feature representation which is beneficial for detection or classification tasks (LeCun et al., 2015). In this study we used deep autoencoder neural networks to construct powerful prediction models for drug-likeness and manually built three larger data sets abstracted from MDDR (MACCS-II Drug Data Report [MDDR], 2004), WDI (Li et al., 2007), ACD (Li et al., 2007) and ZINC (Irwin et al., 2012; Sterling and Irwin, 2015). The molecular descriptors of compound were calculated by Mold2 (Hong et al., 2008) and Padel (Yap, 2011). The classification accuracies of drug-like/non-drug-like model are 91.04% on WDI / ACD databases, and 91.20% on MDDR /ZINC, respectively. The performance of the models outperforms previously reported models. In addition, we developed a drug/non-drug-like model (ZINC World Drug vs. MDDR), which distinguishes drugs and common compounds, with a classification ACC of 96.99%. Our work shows that

TABLE 1 | Detailed information of the dataset pairs.

Dataset pair	Number of positive	Number of negative	Total
WDI/ACD	38,260	288,540	326,800
MDDR/ZINC	171,850	199,220	371,070
WORLDDRUG/ZINC	3,380	199,220	202,600

TABLE 2 | Data preprocessing and post-processing steps used in this study.

Data processing	
Step Name/ Software	**Step description**
Element filter/ KNIME (Berthold et al., 2009)	Hydrocarbons are removed. Molecules containing elements other than C H O N P S Cl Br I Si are removed.
Remove Mixture/ KNIME (Berthold et al., 2009)	All records containing more than one molecules are removed.
Standardize/ ChemAxon Standardizer (ChemAxon Standardizer, 2010)	Neutralize, tautomerize, aromatize, and clean 2D
Remove duplicate / OpenBabel (O'Boyle et al., 2011)	Two molecules having the same InChI(including stereochemistry) means duplication. If a molecule appears in both drug set and nondrug set, it is removed from nondrug set. As for duplications in the same set, only the one that appears first is kept.
Data post-processing	
Remove error values / Python	If a descriptor has the value of N/A or 'infinity', the molecule it belongs to is removed.
Remove constant descriptors / Python	If a descriptor has the same value across all molecules, the descriptor is removed from the descriptor list.

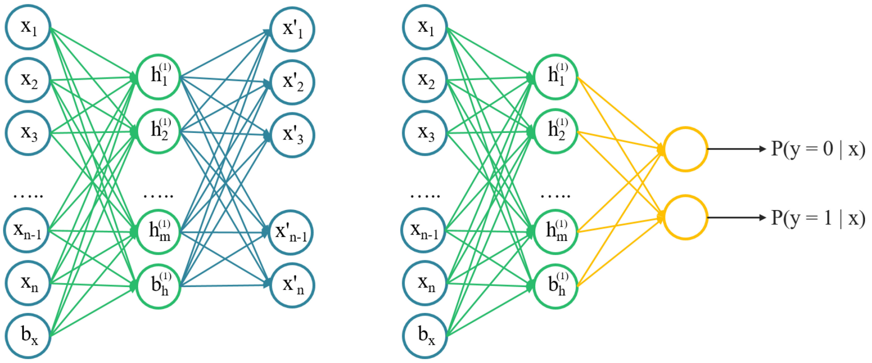


FIGURE 1 | A schematic architecture of a stacked autoencoder. Left) the architecture of autoencoder, layer-by-layer can be stacked. Right) a pre-trained autoencoder to initialize a fully connected network with the same structure for classifying.

by using high-latitude molecular descriptors, we can apply DL technology to establish state-of-the-art drug-likeness prediction models.

Datasets

Benchmark Datasets

In this study, the whole chemical space was divided into drug, drug-like and non-drug-like. Marketed drug molecules were represented by ZINC WORLD DRUG (Sterling and Irwin, 2015) (version 2015, 2500 molecules) dataset. Drug-like molecules were represented by MDDR (MACCS-II Drug Data Report [MDDR], 2004) (200 k molecules) dataset and WDI (Li et al., 2007) (version 2002, 40k molecules) dataset. Non-drug-like molecules were represented by ACD (Li et al., 2007) (version 2002, 300 k molecules) and ZINC ALL PURCHASABLE (Irwin et al., 2012) (version 2012) datasets; the latter was randomly sampled to reduce its size to 200 k. Originally, drug-like datasets contained both marketed and drug-like molecules, and non-drug-like datasets contained the other two datasets. All datasets contained 2D molecular structure information in SDF format. Detailed information of the dataset pairs used in this study can be found in **Table 1**.

Data Preprocessing

Data cleaning can be a crucial step in cheminformatics calculation, as expounded by Fourches et al. (2010). We used a process (see **Table 2**) similar to that of Fourches et al. to preprocess our raw data downloaded, making it less error-prone in descriptor calculation. After descriptor calculations, we also post-processed the resulting descriptor matrix (see **Table 2**).

Descriptor Calculation

We used 2D descriptors to encode the molecules. Molecules after preprocessing were calculated by MOLD2 (Hong et al., 2008), resulting a descriptor matrix of ~700 descriptors per molecule. Then descriptor matrix was subjected to post-processing described in **Table 2**. We also tried the Padel descriptors (Yap, 2011), which showed inferior performance in this study and was discarded.

TABLE 3 | Hyper-parameter settings of the stacked autoencoder.

Hyperparameter	Setting
Initializer	TruncatedNormal
Number of hidden layers	1
Number of hidden layer nodes	512
L2 Normalization term	1e-4
Dropout rate	0.14
Activation	Relu
Batch size	128
Optimizer	Adam
Loss	mse for AE, binary crossentropy for classifier

Over-Sampling Algorithms

Due to the special classification task, the positive and negative samples collected by us were not balanced in this study. Predictive model developed using imbalanced data could be biased and inaccurate. Therefore, we adopted two methods to balance our data sets to make the ratio of positive and negative samples approximately equal. The first method was to copy the minority class making the ratio 1:1, the second one was to use SMOTE (Chawla et al., 2002; Han et al., 2005; Nguyen et al., 2011), which is an improved scheme based on random oversampling algorithm. Here we used imbalanced-learn package downloaded from[1] to apply SMOTE. For each task, we used these two oversampling methods to balance the data. For each model, firstly, we randomly split the datasets on the proportion of 9:1 as training set and validation set, secondly, we used the above two methods to balance the training set, so that the number of positive and negative samples during training was equal. The training set was used to train models with 5-CV and the additional validation set was used to evaluate models.

[1] http://contrib.scikit-learn.org/imbalanced-learn/stable/install.html

TABLE 4 | Performance on the training sets with 5-CV.

Model	Copy the minority class				SMOTE over-sampling			
	ACC	SE	SP	AUC	ACC	SE	SP	AUC
WDI/ACD	0.8923	0.8991	0.8859	0.9598	0.9265	0.9244	0.9286	0.9783
MDDR/ZINC	0.9095	0.8855	0.9302	0.9701	0.9116	0.9141	0.9092	0.9719
WORLD/ZINC	0.9910	0.9961	0.9859	0.9986	0.9906	0.9937	0.9874	0.9990

TABLE 5 | Performance of the models on the validation sets.

Model	Using SMOTE over-sampling				
	ACC	SE	SP	MCC	AUC
WDI/ACD	0.9014	0.7683	0.9191	0.6014	0.9271
MDDR/ZINC	0.9025	0.9012	0.9036	0.8043	0.9669
WORLD/ZINC	0.9800	0.7544	0.9838	0.5690	0.9707

MATERIALS AND METHODS

Stacked Autoencoder

An autoencoder was an unsupervised learning algorithm that trains a neural network to reconstruct its input and more capable of catching the intrinsic structures of input data, instead of just memorizing. Intuitively, it attempted to build an encoding-decoding process so that the output $\hat{x}$ of the model is approximately similar to the input x. The SAE was a neural network consisting of multiple layers of sparse autoencoders, where the output of each layer was connected to the inputs of the successive layer. A schematic architecture of a SAE was shown in **Figure 1**. We trained the AE model with 2D chemical descriptors to find the intrinsic relationship between descriptors, then used the parameters of the AE model to initialize the classification model.

Defining Models

According to the partition of chemical space into drug, drug-like and non-drug-like, there can be two kinds of classification models, drug-like/non-drug-like, drug/non-drug-like. The first one matched the traditional definition of drug-likeness. The second one also bore considerable practical value, but no model had been published to address it. In this study, to address drug-like/non-drug-like classification, we proposed two models, MDDRWDI/ZINC (which means MDDR and WDI as positive set, ZINC as negative set) and WDI/ACD. To address drug/non-drug-like classification, we proposed WORLDDRUG/ZINC (which means ZINC WORLD DRUG as positive set, ZINC ALL PURCHASABLE as negative set) model.

Network Training and Hyperparameter Optimization

In this study, we used the open-source software library Keras (Chollet, 2015) based on Tensorflow (Abadi et al., 2016) to construct SAE model and classification model. Firstly, a single hidden layer AE was trained. The number of hidden layer nodes K, was a hyperparameter needs to be compared

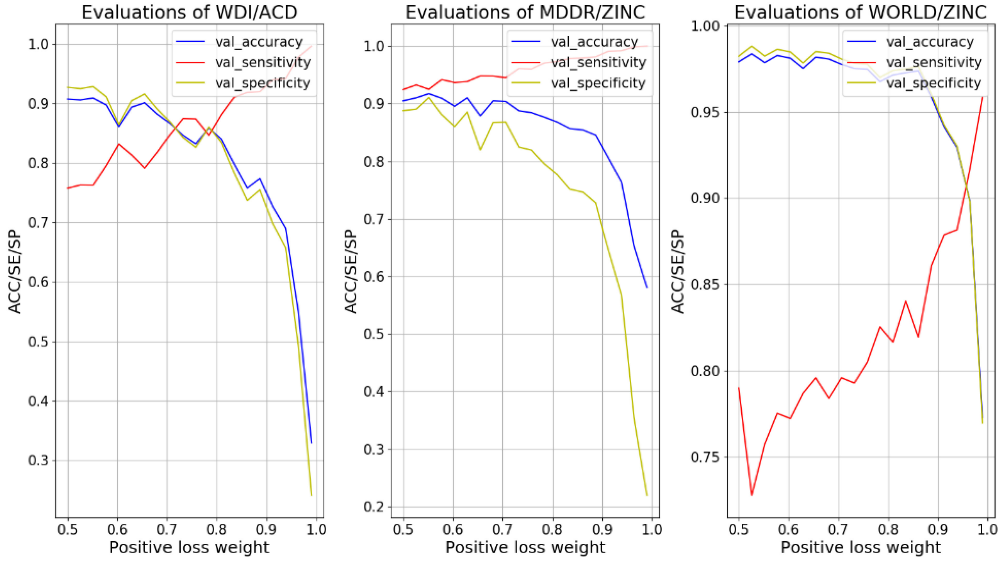


FIGURE 2 | Evaluations of different models vary with weight of positive sample loss.

across different networks and tuned. During training, we used Truncated-Normal initializer to generates a truncated normal distribution of layer weights. In all case, we applied Bayesian optimization (Hyperas, a python library based on hyperopt[2]) to optimize the hyperparameter, such as the number of hidden layer nodes K, the value of L2 weight regularizer, the value of dropout, the type of activation function, the type of optimizer, the value of batch size. The final optimal hyper-parameter settings were listed in **Table 3**.

Considering that although the data set has been balanced, the model results may be overfitting, so we optimized the weight of the positive and negative sample loss of the logarithmic likelihood loss function as:

$$L = -\sum_{k=1}^{n}(wy_k(\log a_k) + (1-w)(1-y_k)\log(1-a_k)) \quad (1)$$

where y_k represented the k$^{\text{th}}$ compound label. $y_k = 1$ or 0, means k$^{\text{th}}$ compound was the drug-like or non-drug-like compound, respectively. $a_k = P(y_k = 1|x_k)$ was the probability to be the drug-like compound of k$^{\text{th}}$ compound calculated by model. w was the weight of the positive sample loss. For different cases, we chose the most suitable w from the range of (0.5∼1.0) to avoid overfitting. Then we trained all models with 5-CV and enforced early stopping based on classification ACC on the test set. Finally, each case had 5 trained models and the average value was the final judgement of these models.

Model Evaluation

All models were evaluated by five indexes. The ACC, SP, and sensitivity(SE), MCC, area under the receiver operating characteristic curve (AUC), the previous four criteria were defined, respectively, as follow:

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \quad (2)$$

$$SP = \frac{TN}{TN + FP} \quad (3)$$

$$SE = \frac{TN}{TP + FN} \quad (4)$$

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(FP + TN)(FP + TP)(FN + TN)(FN + TP)}} \quad (5)$$

RESULTS

Compare Different Over-Sampling Methods

After we tried pre-training on validation test with 5-CV, we found that more layers and neuron numbers did not improve the predictive power. In all case, one hidden layer was sufficient for our classification objective. By analyzing the two different over-sampling methods to balance datasets, copy the minority class and SMOTE, we found the latter can achieve better prediction accuracies in **Table 4**.

TABLE 6 | Performance on the training set after optimizing the weight of loss function.

Model	SMOTE over-sampling				
	ACC	SE	SP	MCC	AUC
WDI/ACD	0.9104	0.9694	0.8515	0.8270	0.9757
MDDR/ZINC	0.9120	0.9219	0.9020	0.8243	0.9726
WORLD/ZINC	0.9699	0.9985	0.9414	0.9416	0.9955

TABLE 7 | Performance on the validation set after optimizing the weight of loss function.

Model	SMOTE over-sampling				
	ACC	SE	SP	MCC	AUC
WDI/ACD	0.8458	0.8524	0.8449	0.5286	0.9253
MDDR/ZINC	0.9046	0.9174	0.8935	0.8095	0.9699
WORLD/ZINC	0.9366	0.8804	0.9376	0.4049	0.9622

With the same dataset, the ACC of a SVM model built by Li et al was 92.73% (Li et al., 2007) and our WDI/ACD model achieves an ACC of 92.65%, almost identical to Li's results. Our MDDRWDI/ZINC model classified drug-like/non-drug-like molecules with a satisfactory ACC of 91.16%, making it the state-of-the-art drug-likeness prediction model. These results suggest that autoencoder is a potential machine learning algorithm in drug-likeness prediction. The ACC of our drug/non-drug-like prediction model based on World Drug/ZINC dataset was as high as 99.06%, showing that it is easier to distinguish compounds from drugs or non-drugs. Although it is not excluded that the ACC of the latter models is related to the serious imbalance of the original data set, we believe that such drug/non-drug-like prediction model will likely benefit drug development.

Optimize the Weights in the Loss Function

We observed that when using the independent external validation set pre-segmented from the original data to evaluate model, the prediction ACC of the model tended to be slightly lower than that of training, but the sensitivity value was significantly lower and the SP value was higher (**Table 5**), indicating that the models have some over-fitting in training.

The underlying reason may be that the positive sample ratio in the original data was too low, and we randomly divided the positive and negative samples in the original data set according to 9:1 to build the training set and the validation set. Even if the SMOTE method was used to balance the positive and negative samples in the train set, the new positive sample generated by SMOTE depended on positive sample in the original training set, so the positive sample information of the external verification set was less included.

In order to overcome the over-fitting on the negative samples, we increased the weight of positive sample loss in the loss

[2] https://github.com/maxpumperla/hyperas

function to enhance the learning ability of the model to the positive sample side. We tested the weigh values (details in Formula 1) from 0.5 to 1 with 20 intervals, and record the values of ACC, SE, and SP on the validation set varying with weight, as shown in **Figure 2**.

For different models, the intersection point of SE and SP in the curves of **Figure 2** corresponded to a balanced weight value. By fine-tuning, the weights corresponding to the four models are (0.69, 0.55 and 0.9). After using these weights for the loss functions, the ACC of the training set in different models fells slightly and the SE improves. As the model reinforces the prediction of positive samples, the SE and SP of the validation set in different models are close (shown in **Tables 6**, **7**).

Although the MCC is generally regarded as a balanced measure, it is seriously affected by the number gap between positive and negative samples of data sets and the confusion matrix calculated by the model. The MCC is satisfactory for the balanced training sets. But in the validation sets, the data set becomes more unbalanced, and the MCC becomes smaller, which was inevitable.

DISCUSSION

In image recognition problems, where AE was originated, several layers of AE are often stacked to make a SAE. Though SAE was found to be more powerful than single layer AE there, we found that SAE is flawed here in drug-likeness problems, making multi-layer SAE perform much poorer than single layer AE.

When a layer of AE is trained, it is expected to give output as close as possible to its input, and the error can be defined as the mean value of output minus input. In this study, when training the model, we found that the ACC of the normalized (z-score) input was much higher than scaling input to [−1,1]. After standardizing the data, the error of AE is 0.8, an order of magnitude higher than typical values in image recognition. Stacking layers of AE will further amplify the error, making the SAE-initialized NN perform poorly in classification.

We propose that such a flaw of AE stems from how input data in different dimensions are interrelated. In image recognition, each pixel is a dimension; in drug-likeness prediction and related areas, each descriptor is a dimension. The training goal of AE is to learn the relationship among dimensions, to encode input information into hidden layer dimensions. So it is very likely that AE would do worse if the relationship among dimensions is intrinsically more chaotic and irregular. The relationship among pixels is regular in that they are organized as a 2D grid and that neighbor pixels bare some similarity and complementarity. Such good properties are absent in relationship among descriptors, resulting in the failure of AE input reconstruction process. Despite the fact that AE reconstruction error is large, our model still performs well in classification. In our opinion, this is due to the regularization effect of AE pre-training. With unsupervised pre-training, the model is more capable of truly learning data, less prone to simply memorizing data.

Imbalanced data sets are a common problem. Although there are some methods such as SMOTE, which can generate new data to balance the data set, this method of generating data is much dependent on the distribution of samples. Once the distribution of samples is very sparse, then the new data is likely to deviate from the space where the original data is exited. Developing method to find data mapping spaces based on the distribution of existing data is critical to generating data to balance the data set, such as the current popular deep generation model. Developing new algorithms to train unbalanced data sets is also an important research direction.

In this study, DL has once again shown its capacity for improving prediction models. Despite the success, we believe that there is still much space for further development. A key aspect is to adapt current DL methods to specific problems. Such adaptations should be based on a better comprehension of current DL methods. That is, knowing which part of the method can be universally applied, and which part should be modified according to the nature of data. For example, in this study, we believe that the regularization effect of AE pre-training is a universal part, while the part of AE input reconstruction should be canceled or modified when input data is irregular.

CONCLUSION

In this study, we manually built two larger data sets, drug-like/non-drug-like and drug/non-drug-like. Then using the AE pre-training method, we developed drug-likeness prediction models. The ACC of classification based on WDI and ACD databases was improved to 91.04%. Our model achieved classification ACC of 91.20% on MDDRWDI/ZINC dataset, making it the state-of-the-art drug-likeness prediction model, showing the predictive power of DL model outperforms traditional machine learning methods. In addition, we developed a drug/non-drug-like model (ZINC World Drug vs. ZINC All Purchasable), which distinguished drugs and common compounds, with a classification ACC of 96.99%. We proposed that AE pre-training served as a better regularization method in this study. The fail of multi-layer SAE reconstruction in this study indicated that due to the specific nature of data, some modifications may be needed when applying DL to different fields. We hope machine learning researchers and chemists collaborate closely to solve such a problem in the future, bringing further comprehension and applications of DL method in chemical problems.

AUTHOR CONTRIBUTIONS

QH and MF wrote the codes and analyzed the data. LL and JP conceived the work. All authors wrote the paper.

FUNDING

This work has been supported in part by the National Natural Science Foundation of China (21673010 and 21633001) and the Ministry of Science and Technology of China (2016YFA0502303).

REFERENCES

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., et al. (2016). "Tensorflow: a system for large-scale machine learning," in *Proceedings of the12th USENIX Symposium on Operating Systems Design and Implementation (OSDI '16)*, Vol. 16 (Savannah, GA: USENIX), 265–283.

Bahdanau, D., Chorowski, J., Serdyuk, D., Brakel, P., and Bengio, Y. (2016). "End-to-end attention-based large vocabulary speech recognition," in *Proceedings of the 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (Lujiazui: IEEE), 4945–4949. doi: 10.1109/ICASSP.2016.7472618

Bengio, Y. (2009). "Learning deep architectures for AI," in *Foundations and Trends® in Machine Learning*, Vol. 2, ed. M. Jordan (Hanover, MA: ACM Digital Library), 1–127.

Berthold, M. R., Cebron, N., Dill, F., Gabriel, T. R., Kötter, T., Meinl, T., et al. (2009). KNIME-the Konstanz information miner: version 2.0 and beyond. *ACM SIGKDD Explor. Newslett.* 11, 26–31. doi: 10.1145/1656274.1656280

Bickerton, G. R., Paolini, G. V., Besnard, J., Muresan, S., and Hopkins, A. L. (2012). Quantifying the chemical beauty of drugs. *Nat. chem.* 4:90. doi: 10.1038/nchem.1243

Byvatov, E., Fechner, U., Sadowski, J., and Schneider, G. (2003). Comparison of support vector machine and artificial neural network systems for drug/nondrug classification. *J. Chem. Inf. Comput. Sci.* 43, 1882–1889. doi: 10.1021/ci0341161

Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. *J. Artif. Intell. Res.* 16, 321–357. doi: 10.1613/jair.953

ChemAxon Standardizer. (2010). *ChemAxon Standardizer Version 5.4.4.1*. Budapest: ChemAxon.

Chen, Y., Li, Y., Narayan, R., Subramanian, A., and Xie, X. (2016). Gene expression inference with deep learning. *Bioinformatics* 32, 1832–1839. doi: 10.1093/bioinformatics/btw074

Chollet, F. (2015). *Keras. GitHub*. Available at: https://github.com/fchollet/keras.

Coates, A., Ng, A., and Lee, H. (2011). "An analysis of single-layer networks in unsupervised feature learning," in *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics* (Fort Lauderdale, FL: PMLR), 215–223.

Dahl, G. E., Yu, D., Deng, L., and Acero, A. (2012). Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition. *IEEE Trans. Audio Speech Lang. Process.* 20, 30–42. doi: 10.1109/TASL.2011.2134090

Darrow, J. J., and Kesselheim, A. S. (2014). Drug development and FDA approval, 1938–2013. *N. Engl. J. Med.* 370:e39. doi: 10.1056/NEJMp1402114

Di Lena, P., Nagata, K., and Baldi, P. (2012). Deep architectures for protein contact map prediction. *Bioinformatics* 28, 2449–2457. doi: 10.1093/bioinformatics/bts475

Fourches, D., Muratov, E., and Tropsha, A. (2010). Trust, but verify: on the importance of chemical structure curation in cheminformatics and QSAR modeling research. *J. Chem. Inf. Comput. Sci.* 50, 1189–1204. doi: 10.1021/ci100176x

Graves, A., Mohamed, A. R., and Hinton, G. (2013). "Speech recognition with deep recurrent neural networks," in *Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (Vancouver: IEEE), 6645–6649. doi: 10.1109/ICASSP.2013.6638947

Han, H., Wang, W. Y., and Mao, B. H. (2005). "Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning," in *Proceedings of the International Conference on Intelligent Computing* (Berlin: Springer), 878–887. doi: 10.1007/11538059_91

He, K., Zhang, X., Ren, S., and Sun, J. (2016). "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Seattle, WA: IEEE), 770–778. doi: 10.1109/CVPR.2016.90

Heffernan, R., Paliwal, K., Lyons, J., Dehzangi, A., Sharma, A., Wang, J., et al. (2015). Improving prediction of secondary structure, local backbone angles, and solvent accessible surface area of proteins by iterative deep learning. *Sci. Rep.* 5:11476. doi: 10.1038/srep11476

Hinton, G. E., Osindero, S., and Teh, Y. W. (2006). A fast learning algorithm for deep belief nets. *Neural Comput.* 18, 1527–1554. doi: 10.1162/neco.2006.18.7.1527

Hong, H., Xie, Q., Ge, W., Qian, F., Fang, H., Shi, L., et al. (2008). Mold2, molecular descriptors from 2D structures for chemoinformatics and toxicoinformatics. *J. Chem. Inf. Model.* 48, 1337–1344. doi: 10.1021/ci800038f

Irwin, J. J., Sterling, T., Mysinger, M. M., Bolstad, E. S., and Coleman, R. G. (2012). ZINC: a free tool to discover chemistry for biology. *J. Chem. Inf. Model.* 52, 1757–1768. doi: 10.1021/ci3001277

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). "Imagenet classification with deep convolutional neural networks," in *Proceedings of the Conference on Advances in Neural Information Processing Systems* (Lake Tahoe, NV: ACM Digital Library), 1097–1105.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature* 521:436. doi: 10.1038/nature14539

Li, Q., Bender, A., Pei, J., and Lai, L. (2007). A large descriptor set and a probabilistic kernel-based classifier significantly improve druglikeness classification. *J. Chem. Inf. Model.* 47, 1776–1786. doi: 10.1021/ci700107y

Lipinski, C. A. (2004). Lead-and drug-like compounds: the rule-of-five revolution. *Drug Discov. Today Technol.* 1, 337–341. doi: 10.1016/j.ddtec.2004.11.007

Lipper, R. A. (1999). How can we optimize selection of drug development candidates from many compounds at the discovery stage? *Mod. Drug Discov.* 2, 55–60.

Lyons, J., Dehzangi, A., Heffernan, R., Sharma, A., Paliwal, K., Sattar, A., et al. (2014). Predicting backbone Cα angles and dihedrals from protein sequences by stacked sparse auto-encoder deep neural network. *J. Comput. Chem.* 35, 2040–2046. doi: 10.1002/jcc.23718

MACCS-II Drug Data Report [MDDR] (2004). *MACCS-II Drug Data Report [MDDR]*. San Leandro, CA: Molecular Design Limited.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). "Distributed representations of words and phrases and their compositionality," in *Proceedings of the conference on Advances in Neural Information Processing Systems* (Lake Tahoe, NV: ACM Digital Library), 3111–3119.

Müller, K. R., Rätsch, G., Sonnenburg, S., Mika, S., Grimm, M., and Heinrich, N. (2005). Classifying 'drug-likeness' with kernel-based learning methods. *J. Chem. Inf. Model.* 45, 249–253. doi: 10.1021/ci049737o

Nguyen, H. M., Cooper, E. W., and Kamei, K. (2011). Borderline over-sampling for imbalanced data classification. *Int. J. Knowl. Eng. Soft Data Paradig.* 3, 4–21. doi: 10.1016/j.jbi.2017.03.002

O'Boyle, N. M., Banck, M., James, C. A., Morley, C., Vandermeersch, T., and Hutchison, G. R. (2011). Open Babel: an open chemical toolbox. *J. Cheminform.* 3:33. doi: 10.1186/1758-2946-3-33

Schneider, N., Jäckels, C., Andres, C., and Hutter, M. C. (2008). Gradual in silico filtering for druglike substances. *J. Chem. Inf. Model.* 48, 613–628. doi: 10.1021/ci700351y

Socher, R., Bengio, Y., and Manning, C. D. (2012). "Deep learning for NLP (without magic)," in *Tutorial Abstracts of ACL 2012*, ed. M. Strube (Stroudsburg, PA: Association for Computational Linguistics), 12–14.

Sterling, T., and Irwin, J. J. (2015). ZINC 15–ligand discovery for everyone. *J. Chem. Inf. Model.* 55, 2324–2337. doi: 10.1021/acs.jcim.5b00559

Tian, S., Wang, J., Li, Y., Xu, X., and Hou, T. (2012). Drug-likeness analysis of traditional chinese medicines: prediction of drug-likeness using machine learning approaches. *Mol. Pharm.* 9, 2875–2886. doi: 10.1021/mp300198d

Venkatesh, S., and Lipper, R. A. (2000). Role of the development scientist in compound lead selection and optimization. *J. Pharm. Sci.* 89, 145–154. doi: 10.1002/(SICI)1520-6017(200002)89:2<145::AID-JPS2>3.0.CO;2-6

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., and Manzagol, P. A. (2010). Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion. *J. Mach. Learn. Res.* 11, 3371–3408.

Wagener, M., and van Geerestein, V. J. (2000). Potential drugs and nondrugs: prediction and identification of important structural features. *J. Chem. Inf. Comput. Sci.* 40, 280–292. doi: 10.1021/ci990266t

Yap, C. W. (2011). PaDEL-descriptor: an open source software to calculate molecular descriptors and fingerprints. *J. Comput. Chem.* 32, 1466–1474. doi: 10.1002/jcc.21707

Zeng, H., Edwards, M. D., Liu, G., and Gifford, D. K. (2016). Convolutional neural network architectures for predicting DNA–protein binding. *Bioinformatics* 32, i121–i127. doi: 10.1093/bioinformatics/btw255

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

frontiers in Genetics

ORIGINAL RESEARCH
published: 04 December 2018
doi: 10.3389/fgene.2018.00587

RDAD: A Machine Learning System to Support Phenotype-Based Rare Disease Diagnosis

Jinmeng Jia[1†], *Ruiyuan Wang*[1†], *Zhongxin An*[1], *Yongli Guo*[2*], *Xi Ni*[2*] *and Tieliu Shi*[1,3*]

[1] The Center for Bioinformatics and Computational Biology, Shanghai Key Laboratory of Regulatory Biology, The Institute of Biomedical Sciences and School of Life Sciences, East China Normal University, Shanghai, China, [2] Beijing Key Laboratory for Pediatric Diseases of Otolaryngology, Head and Neck Surgery, The Ministry of Education Key Laboratory of Major Diseases in Children, Beijing Pediatric Research Institute, Beijing Children's Hospital, Capital Medical University, National Center for Children's Health, Beijing, China, [3] National Center for International Research of Biological Targeting Diagnosis and Therapy/Guangxi Key Laboratory of Biological Targeting Diagnosis and Therapy Research/Collaborative Innovation Center for Targeting Tumor Diagnosis and Therapy, Guangxi Medical University, Nanning, Guangxi, China

OPEN ACCESS

Edited by:
Minjun Chen,
National Center for Toxicological Research (FDA), United States

Reviewed by:
Youping Deng,
Rush University Medical Center, United States
Wenzhong Xiao,
Harvard Medical School, United States

***Correspondence:**
Yongli Guo
guoyongli@bch.com.cn
Xi Ni
nixin@bch.com.cn
Tieliu Shi
tlshi@bio.ecnu.edu.cn

†These authors have contributed equally to this work

Specialty section:
This article was submitted to Bioinformatics and Computational Biology, a section of the journal Frontiers in Genetics

Received: *12 August 2018*
Accepted: *15 November 2018*
Published: *04 December 2018*

Citation:
Jia J, Wang R, An Z, Guo Y, Ni X and Shi T (2018) RDAD: A Machine Learning System to Support Phenotype-Based Rare Disease Diagnosis. Front. Genet. 9:587. doi: 10.3389/fgene.2018.00587

DNA sequencing has allowed for the discovery of the genetic cause for a considerable number of diseases, paving the way for new disease diagnostics. However, due to the lack of clinical samples and records, the molecular cause for rare diseases is always hard to identify, significantly limiting the number of rare Mendelian diseases diagnosed through sequencing technologies. Clinical phenotype information therefore becomes a major resource to diagnose rare diseases. In this article, we adopted both a phenotypic similarity method and a machine learning method to build four diagnostic models to support rare disease diagnosis. All the diagnostic models were validated using the real medical records from RAMEDIS. Each model provides a list of the top 10 candidate diseases as the prediction outcome and the results showed that all models had a high diagnostic precision (≥98%) with the highest recall reaching up to 95% while the models with machine learning methods showed the best performance. To promote effective diagnosis for rare disease in clinical application, we developed the phenotype-based Rare Disease Auxiliary Diagnosis system (RDAD) to assist clinicians in diagnosing rare diseases with the above four diagnostic models. The system is freely accessible through http://www.unimd.org/RDAD/.

Keywords: rare disease, phenotype, machine learning, diagnostic model, web-based tools

INTRODUCTION

Rare diseases are rare conditions that occur only in a precious few people. Currently, there is no unified, widely accepted definition for rare diseases (Jia and Shi, 2017). To facilitate increased communication, knowledge sharing and coordinated orphan drug development across national borders, the World Health Organization (WHO) defines rare diseases as a prevalence >6.5–10 in 10,000 (Franco, 2013), which we adopted as the definition of rare diseases in this article. About 80% of rare diseases are the consequence of genetic defects, but >5% of rare diseases can be effectively interfered with or treated. Nowadays, screening and diagnostic rates of rare diseases are constantly improved with the progress of molecular biology and cytogenetics (Ekins, 2017). For example, whole-exome sequencing has allowed for the discovery of the genetic cause for a considerable number of diseases, opening up new ways for disease diagnostics, especially for OMIM (Online

Mendelian Inheritance in Man) disorders. However, due to the lack of clinical samples and records, the molecular causeremains difficult to identify (Qi et al., 2017; Wu et al., 2017). Therefore, only a limited number of rare Mendelian diseases can be diagnosed through DNA sequencing, making clinical phenomic information a major resource to diagnose rare diseases (Jia and Shi, 2017). Disease phenotypes (also known as clinical phenotypes) refer to the observable characteristics of an organism (or cell), including individual form, function and other aspects of performance, such as height, color, blood type and enzyme activity. Usually, phenotypes associated with rare diseases are described by a set of clinical medical terms. To provide better interoperability in the field of rare diseases, several tools have been specifically designed to assist in standardizing, and sharing of clinical medical terms, through various medical resources (Dragusin et al., 2013; Girdea et al., 2013; Yang et al., 2015; Maiella et al., 2018). For example, Phenomizer aims to help diagnose genetic diseases from the input list of symptoms and PhenoTips provides a framework to share and analyze patient data between professionals. At present, the main approach to support disease diagnosis is based on disease similarities calculated from diseases' clinical phenotypes, using a semantic hierarchy of the Human Phenotype Ontology (HPO; Alves et al., 2016). Under this circumstance, the similarity score between two diseases will be highly dependent on the completeness and specificity of their annotated phenotypes. To overcome the limitations, we adopted both the traditional phenotypic similarity method and a new machine learning method to build four diagnostic models to support the diagnosis of rare diseases. We then validated the performance of all these models using the real electronic medical records (EMR) from RAMEDIS. To promote effective diagnosis of rare diseases in a clinical application, we developed the phenotype-based Rare Disease Auxiliary Diagnosis system (RDAD) to assist clinicians in diagnosing rare diseases using the above four diagnostic models.

MATERIALS AND METHODS

The workflow of the RDAD is depicted in **Figure 1**. The data sets for the four diagnostic models contained in the RDAD were integrated from eRAM (Jia et al., 2018a), Human Phenotype Ontology (Robinson et al., 2008), Orphanet (Pavan et al., 2017), OMIM (Amberger et al., 2014), and DECIPHER (Firth et al., 2009). To integrate multi-level biomedical resources and multiple classifiers, we built four diagnostic models. The phenotype based rare disease similarity (PICS) model used curated rare disease-phenotype associations as the input data and four disease similarity methods as the classifiers, while the phenotype-gene based rare disease similarity (PGAS) model used curated rare disease-phenotype associations and curated phenotype-gene associations as the input data and two disease similarity methods as the classifiers. In contrast, the phenotype based machine learning (CPML) model used curated rare disease-phenotype associations as the input data and six machine learning algorithms as the classifiers; similarly, the curated and text-mined phenotype based machine learning (APML) model used curated rare disease-phenotype associations and text mined (Xu et al., 2013) rare disease-phenotype associations as the input data and six machine learning algorithms as the classifiers. The four different diagnostic models contained in the RDAD system, with their input data sets and classifiers are listed in **Table 1**.

Extraction of Phenotypes and Corresponding Genes

Rare disease names were extracted from eRAM, the rare disease-phenotype associations were extracted from HPO and eRAM, rare disease related genes were mainly collected from eRAM. eRAM is a standardized system that covers a variety of rare diseases, integrates current existing data on clinical manifestations (symptoms and phenotypes) and molecular mechanisms of rare diseases systematically, revealing many novel associations between rare diseases (Jia et al., 2018a). The HPO is a system providig a standardized vocabulary of phenotypic abnormalities that are encountered in human disease (Robinson et al., 2008). We first obtained rare disease names from eRAM, then extracted curated rare disease-phenotype associations from the annotation files (phenotype_annotation_hpoteam.tab, #1249) provided by HPO, which contains annotations made explicitly and manually by the HPO-team (mostly referring to OMIM entries). In addition, we retrieved the rare disease-phenotype pairs from eRAM in which the related records were extracted from abstracts and full-text articles in MEDLINE, through a pattern-based relationship extraction approach (Xu et al., 2013). In total, 8,488,796 abstracts and 774,514 full-text articles were text-mined, respectively, from PubMed and PubMed Central, which lead to the identification of 23,231 rare disease-phenotype pairs.

Electronic Health Records

RAMEDIS (Rare Metabolic Diseases Database) provides an accurate curated resource of human variations with corresponding phenotypes for rare metabolic diseases (Topel et al., 2010). So far, 93 different genetic metabolic diseases among 818 patients have been released. PhenoTips is an open source framework for analyzing phenotype information for patients with genetic diseases (Girdea et al., 2013). We downloaded all 1,099 medical records from RAMEDIS, and then obtained 818 related records according to the mapping between the diagnostic disease names of medical records (rare disease names were standardized by eRAM). According to the historical description and symptom fields in the medical records, the corresponding phenotypic data from the medical records were extracted with the open source software PhenoTips. Finally, 309 phenotypes were obtained, involving 27 rare diseases, which were subsequently used as the real medical records-based test set for the four different diagnosis models contained in RDAD. The test set extracted from RAMEDIS is listed in **Table 2**.

The PICS Diagnostic Model

The input data of the PICS diagnostic model were the curated rare disease-phenotype associations, and we selected the curated phenotypes as the features. Cosine similarity is defined as the evaluation of the similarity between two vectors by calculating

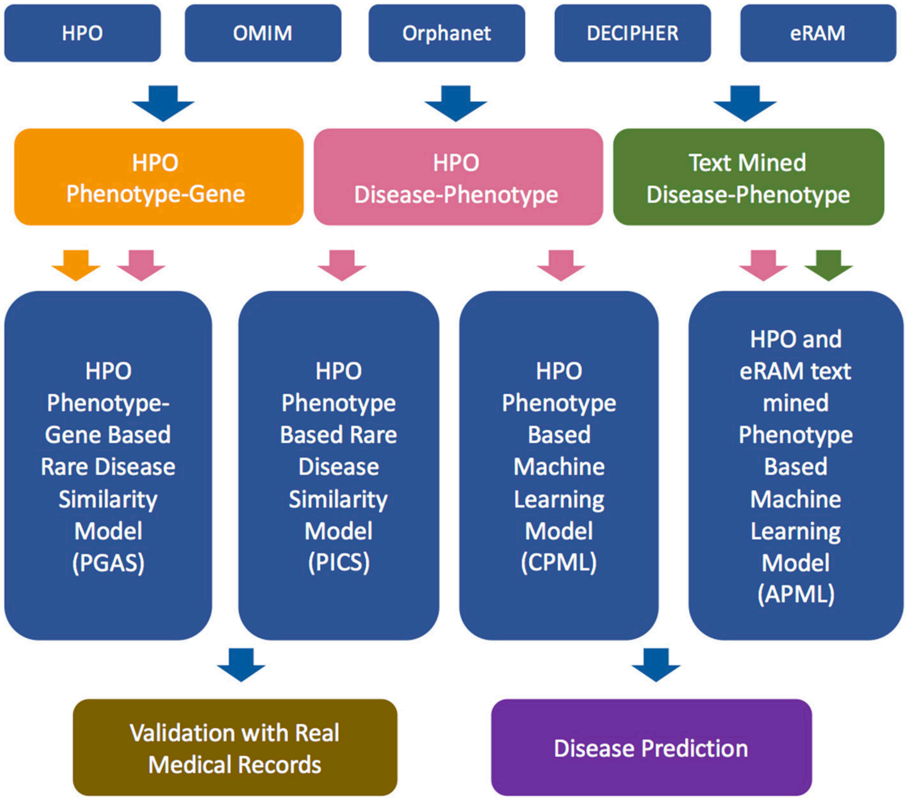

HPO
OMIM
Orphanet
DECIPHER
eRAM
HPO Phenotype-Gene
HPO Disease-Phenotype
Text Mined Disease-Phenotype
HPO Phenotype-Gene Based Rare Disease Similarity Model (PGAS)
HPO Phenotype Based Rare Disease Similarity Model (PICS)
HPO Phenotype Based Machine Learning Model (CPML)
HPO and eRAM text mined Phenotype Based Machine Learning Model (APML)
Validation with Real Medical Records
Disease Prediction

FIGURE 1 | The workflow of RDAD. HPO, Human Phenotype Ontology. OMIM, Online Mendelian Inheritance in Man. PGAS, Phenotype-Gene Association based rare disease similarity model; PICS, Phenotypic TF-IDF-Hierarchy information content based rare disease similarity model; CPML, Curated feature Phenotype spatial vector based rare disease Machine Learning prediction model; APML, Curated and text mined feature phenotype spatial vector based rare disease Machine Learning prediction model.

TABLE 1 | The Four Diagnostic Models Contained in the RDAD System.

Data sources	**Model**			
	PICS	**PGAS**	**CPML**	**APML**
HPO Phenotypes	√	√	√	√
eRAM Curated Genes		√		
eRAM Text Mined Phenotypes				√
Disease Similarity Classifiers	√	√		
Machine Learning Classifiers			√	√

the value of the angle cosine. The similarity between the two vectors of the same vector cosine is 1, and the similarity of the two vectors at 90 degrees is 0. If the two vectors are the opposite, the similarity is −1. Cosine similarity is used in the positive space and the value is to be neatly bound in [0,1] (Jia et al., 2018b). Given two feature phenotype spatial vectors, $D = (p_1, p_2, \ldots, p_n)$, $Q = (q_1, q_2, \ldots, q_n)$, the cosine similarity is represented using a dot product and magnitude as follows:

$$Cosine\_similarity = \frac{D{*}Q}{\parallel D \parallel * \parallel Q \parallel}$$

$$= \frac{\sum_{i=1}^{n} (D_i{*}Q_i)}{\sqrt{\sum_{i=1}^{n} (D_i)^2}{*}\sqrt{\sum_{i=1}^{n} (Q_i)^2}}$$

The Tanimoto coefficient is extended by the Jaccard coefficient. Given two feature phenotype spatial vectors, $D = (p_1, p_2, \ldots, p_n)$, $Q = (q_1, q_2, \ldots, q_n)$, the Tanimoto coefficient is calculated as follows:

$$Tanimoto(D, Q) = \frac{D{*}Q}{\parallel D \parallel^2 + \parallel Q \parallel^2 - D{*}Q}$$

To provide an antidiastole and to rank the candidate rare diseases in descending order of probability, the score is calculated as follows (Pinol et al., 2017):

$$\Psi_i = 1 - \frac{n}{Max[P_u, P_i]}$$

Where P_u indicates the phenotypes provided by the user, P_i indicates the phenotypes of rare diseases in the training set, the function of $Max[P_u, P_i]$ refers to the largest number between P_u and P_i. n signifies the number of different phenotypes between the phenotypes associated with any rare disease in the RDAD database and the phenotypes submitted by the user.

TABLE 2 | The Test Data Set for the Four Diagnostic Models.

Diagnosis	Case count
PHENYLKETONURIA (MIM 261600)	157
CONGENITAL DISORDER OF GLYCOSYLATION, TYPE Ia (MIM 212065)	27
MAPLE SYRUP URINE DISEASE (MIM 248600)	21
PROPIONIC ACIDEMIA (MIM 606054)	16
CANAVAN DISEASE (MIM 271900)	15
SUCCINIC SEMIALDEHYDE DEHYDROGENASE DEFICIENCY (MIM 271980)	10
ALKAPTONURIA (MIM 203500)	10
ARGININOSUCCINIC ACIDURIA (MIM 207900)	9
ISOVALERIC ACIDEMIA (MIM 243500)	7
CYSTINURIA (MIM 220100)	5
CITRULLINEMIA, TYPE II, NEONATAL-ONSET (MIM 605814)	5
WILSON DISEASE (MIM 277900)	4
HOLOCARBOXYLASE SYNTHETASE DEFICIENCY (MIM 253270)	4
FANCONI-BICKEL SYNDROME (MIM 227810)	2
ALPHA-METHYLACETOACETIC ACIDURIA (MIM 203750)	2
TYROSINE TRANSAMINASE DEFICIENCY (MIM 276600)	2
HYPERINSULINEMIC HYPOGLYCEMIA, FAMILIAL, 2 (MIM 601820)	2
HAWKINSINURIA (MIM 140350)	2
OSTEOGENESIS IMPERFECTA, TYPE I (MIM 166200)	1
GLYCOGEN STORAGE DISEASE VI (MIM 232700)	1
N-ACETYLGLUTAMATE SYNTHASE DEFICIENCY (MIM 237310)	1
REFSUM DISEASE (MIM 266500)	1
KRABBE DISEASE (MIM 245200)	1
LEIGH SYNDROME (MIM 256000)	1
GLYCOGEN STORAGE DISEASE Ib (MIM 232220)	1
PYRUVATE CARBOXYLASE DEFICIENCY (MIM 266150)	1
PEARSON MARROW-PANCREAS SYNDROME (MIM 557000)	1

The similarity between two phenotypes can be calculated by the "information content" of their MICA (Most Informative Common Ancestor; Kohler et al., 2009). For each of the phenotypes submitted by the user, the best matched phenotype among the phenotypes related to the rare disease is found, and the average value over all the query phenotypes is then calculated. The similarity is calculated as follows:

$$Similarity\left(Q \rightarrow D\right) = avg\left[\sum_{p_1 \in Q} \max_{p_2 \in D} IC\left(MICA\left(p_1, p_2\right)\right)\right]$$

The symmetric version of the above equation is:

$$\begin{aligned} Similarity_{symmetric}\left(D, Q\right) = & \frac{1}{2} Similarity\left(Q \rightarrow D\right) \\ & + \frac{1}{2} Similarity\left(D \rightarrow Q\right) \end{aligned}$$

Based on the TF-IDF-Hierarchy information content (van Driel et al., 2006) matrix of rare disease associated phenotype spatial vector obtained from Data Set I, we used the above methods to construct the PICS model.

The PGAS Diagnostic Model

The input data of the PGAS diagnostic model were the curated rare disease-phenotype associations and the curated phenotype-gene associations, and we selected the curated genes and curated phenotypes as the features.

Given two feature gene spatial vectors, $G = (g_1, g_2, \ldots, g_n)$, $Q = (q_1, q_2, \ldots, q_n)$, the cosine similarity, is represented using a dot product as follows:

$$\begin{aligned} Cosine\_similarity = & \frac{G{*}Q}{\parallel G \parallel * \parallel Q \parallel} \\ = & \frac{\sum_{i=1}^{n}\left(G_i{*}Q_i\right)}{\sqrt{\sum_{i=1}^{n}\left(G_i\right)^2}{*}\sqrt{\sum_{i=1}^{n}\left(Q_i\right)^2}} \end{aligned}$$

Given two feature gene spatial vectors, $G = (g_1, g_2, \ldots, g_n)$, $Q = (q_1, q_2, \ldots, q_n)$, the Tanimoto coefficient, is represented as follows:

$$Tanimoto(G, Q) = \frac{G{*}Q}{\parallel G \parallel^2 + \parallel Q \parallel^2 - G{*}Q}$$

Given two phenotype sets, $P_1 = (p_1, p_2, \ldots, p_m)$, $P_2 = (p_1, p_2, \ldots, p_n)$, the similarities between two phenotype sets are defined as follows:

$$\begin{aligned} Similarity_{symmetric}\left(P_1, P_2\right) = & \frac{1}{2} Similarity\left(P_1 \rightarrow P_2\right) \\ & + \frac{1}{2} Similarity\left(P_2 \rightarrow P_1\right) \end{aligned}$$

Based on the rare disease associated phenotype-gene spatial vector obtained from Data Set II, we used the above methods to construct the PGAS model.

The CPML Diagnostic Model and the APML Diagnostic Model

The input data of the CPML diagnostic model were the curated rare disease-phenotype associations, and we selected the curated phenotypes as the features. Similarly, the input data of the APML diagnostic model were the curated rare disease-phenotype associations and the text mined rare disease-phenotype associations, and we selected the curated phenotypes and text mined phenotypes as the features.

Based on the TF-IDF-Hierarchy information content matrix of rare disease associated phenotype spatial vector obtained from Data Set III and Data Set IV, the CPML model, and the APML model take classifier performance into consideration. We first adopted Logistic Regression, KNN, Random Forest, Extra Trees, Naive Bayes, and Deep Neural Network machine learning classification algorithms as classifiers, respectively, and then used the Bayesian averaging algorithm in both models to leverage the prediction results of the six classifiers, ranking candidate rare diseases by their scores.

The Data Sets for the Four Diagnostic Models Contained in the RDAD System

The training sets for the four diagnostic models contained in the RDAD system are listed in **Table 3**. All rare diseases in the four training data sets were regarded as model labels. The phenotypes in Data Set I/III/VI were used to calculate the phenotypic TF-IDF-Hierarchy information content, based on the phenotype semantic hierarchy of HPO. The genes in Data Set II were used to calculate phenotype similarity and the phenotypes in Data Set II were used to calculate the rare disease similarity based on the phenotype similarity in the PGAS model. The records in Data III/IV were used as the input data for the machine learning classifiers in the CPML model and the APML model.

The Four Diagnostic Models in the RDAD System With Their Corresponding Classifiers

To facilitate rare disease diagnosis, we applied the phenotypic TF-IDF-Hierarchy information content on the phenotype semantic hierarchy of Human Phenotype Ontology (HPO), and then built the phenotypic TF-IDF-Hierarchy information content based on the rare disease similarity model (PICS), the phenotype-gene association based rare disease similarity model (PGAS), and the curated feature phenotype spatial vector based rare disease machine learning prediction model (CPML), as well as the curated and text mined feature phenotype spatial vector based rare disease machine learning prediction model (APML). The four diagnostic models contained in RDAD with their corresponding classifiers are listed in **Table 4**.

Precision and Recall

Precision measures the fraction of correct predictions made by the four diagnostic models contained in the RDAD system. Recall (or specificity) measures the fraction calculated by dividing the number of correct choices by the total number of choices available to each model. True positives (TP) are the number of correctly predicted rare diseases, false positives (FP) are the number of incorrectly predicted rare diseases and false negatives (FN) are the number of rare diseases that are not predicted. The F1-score is an aggregate measure for the accuracy of a classifier that calculates a weighted average of Precision and Recall defined as follows (Alves et al., 2016):

$$\begin{aligned} Precision &= \frac{TP}{TP + FP} \\ Recall &= \frac{TP}{TP + FN} \\ F1\_Score &= \frac{Precision * Recall}{2 * \left(Precision + Recall\right)} \end{aligned}$$

RESULTS

Precision and Recall

We validated the above four models with the real medical records from RAMEDIS. The results showed that the PICS model achieved the best performance among the four models, with only one rare disease as the outcome of the prediction (**Figure 2A**), but in real application, the diagnosis result is barely satisfactory. To better help clinicians pinpoint the

TABLE 3 | The Training Data Sets for the Four Diagnostic Models.

Data set	Model	Term	Term count
Data Set I	PICS	Rare Diseases	4,498
		Curated Phenotypes	5,990
		D-P Associations	57,346
Data Set II	PGAS	Rare Diseases	4,498
		Curated Phenotypes	5,990
		D-P Associations	57,346
		Curated Genes	3,682
		P-G Associations	419,597
Data Set III	CPML	Rare Diseases	4,498
		Curated Phenotypes	5,990
		D-P Associations	57,346
		Synthetic Patients	44,980
Data Set IV	APML	Rare Diseases	4,498
		All Phenotypes	6,453
		D-P Associations	72,404
		Synthetic Patients	44,980

D-P Associations, Disease-Phenotype association pairs; P-G Associations, Phenotype-Gene association pairs.

TABLE 4 | The Four Diagnostic Models with Their Corresponding Classifiers.

Model	Data set	Classifier	Score
PICS	Data Set I	Cosine Similarity Tanimoto Coefficient Ψ_i Score MICA	Bayesian Averaging Algorithm
PGAS	Data Set II	Cosine Similarity Tanimoto Coefficient	Bayesian Averaging Algorithm
CPML	Data Set III	Logistic Regression K-Nearest Neighbor Random Forest Extra Trees Naive Bayes Deep Neural Network	Bayesian Averaging Algorithm
APML	Data Set IV	Logistic Regression K-Nearest Neighbor Random Forest Extra Trees Naive Bayes Deep Neural Network	Bayesian Averaging Algorithm

MICA, Most Informative Common Ancestor.

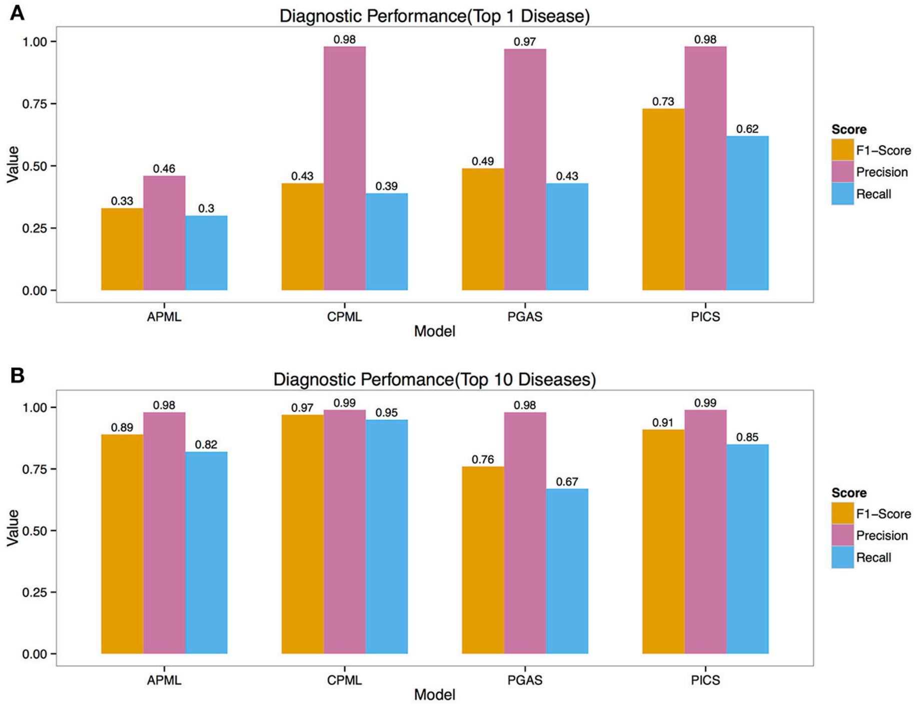


FIGURE 2 | The Precision, Recall, F1-Score of Different Models. **(A)** The top 1 diagnostic performance. **(B)** The top 10 diagnostic performance. APML, the curated and text mined feature phenotype spatial vector based rare disease machine learning prediction model. CPML, the curated feature phenotype spatial vector based rare disease machine learning prediction model. PGAS, the phenotype-gene association based rare disease similarity model. PICS, the phenotypic TF-IDF-Hierarchy information content based rare disease similarity model.

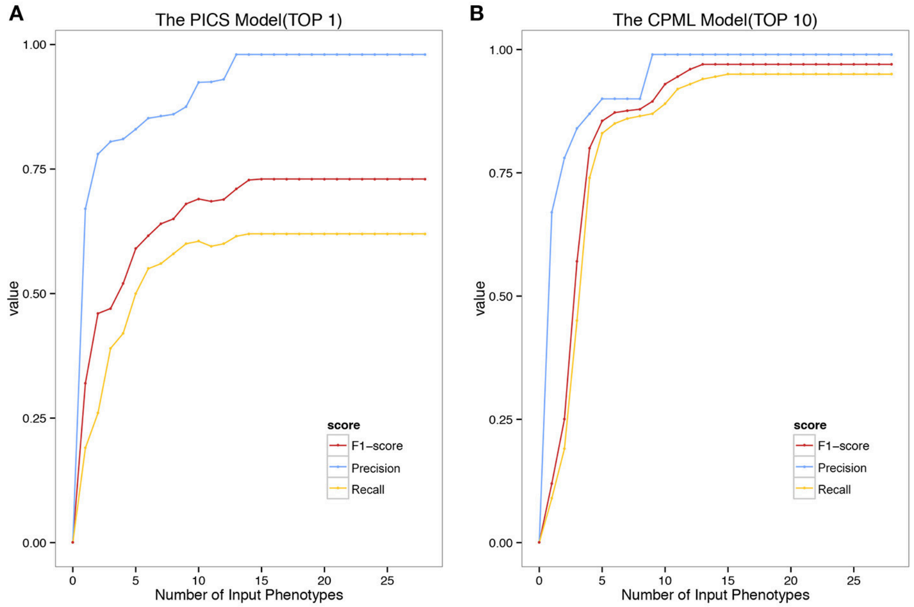


FIGURE 3 | The Precision Recall and F1-Score of the model with different number of Phenotypes Submitted. **(A)** The top 1 diagnostic performance of PICS model. **(B)** The top 10 diagnostic performance of CPML model.

right disease, we then provided a credible list of the top 10 candidate diseases as the prediction outcome, which will help clinicians narrow down candidate diseases through the diagnostic process. Under such circumstances, the CPML model had the best performance (**Figure 2B**). In addition, in order to achieve the best result for rare disease diagnosis, RDAD suggests that the number of inputted phenotype terms of each selected diagnostic model is around 15 (**Figure 3**). The average number of symptoms recorded in the EMR in RAMEDIS database was 17, indicating that the suggested number of the RDAD model (around 15) is feasible.

Confusion Matrix

The confusion matrix is a special two-dimensional contingency table with the same class set on two dimensions. We built a confusion matrix of the top 10 rare disease candidates for each model using the EMR from RAMEDIS. The confusion matrixes of different models showed that machine learning diagnostic models (CPML and APML) performed better than traditional disease similarity models (PICS and PGAS). Compared to other models, the CPML model showed the best performance (**Figure 4**, **Figure S1**).

Candidate Rare Diseases Rank

Given the input phenotypes, we examined the candidate rare diseases detected, ranked as top 1, top 10 and others with the four diagnostic models in RDAD. We found that 62.1% of the designated rare diseases were ranked as top 1 with the PICS model, the good performance of this model is most likely due to the accuracy of the associated phenotype of rare diseases and the direct calculation between the spatial vectors while the other three models undergo a series of transformations during data processing, resulting in information loss and error amplification. In contrast, 95.5% of the correct rare diseases were ranked as top 10 with the CPML model. Thus, our results clearly demonstrate that the four diagnostic models contained in the RDAD system are suitable for finding rare diseases that are known to be associated with phenotypes. In general, the model built by the machine learning method, showed better performance. The four diagnostic models successfully

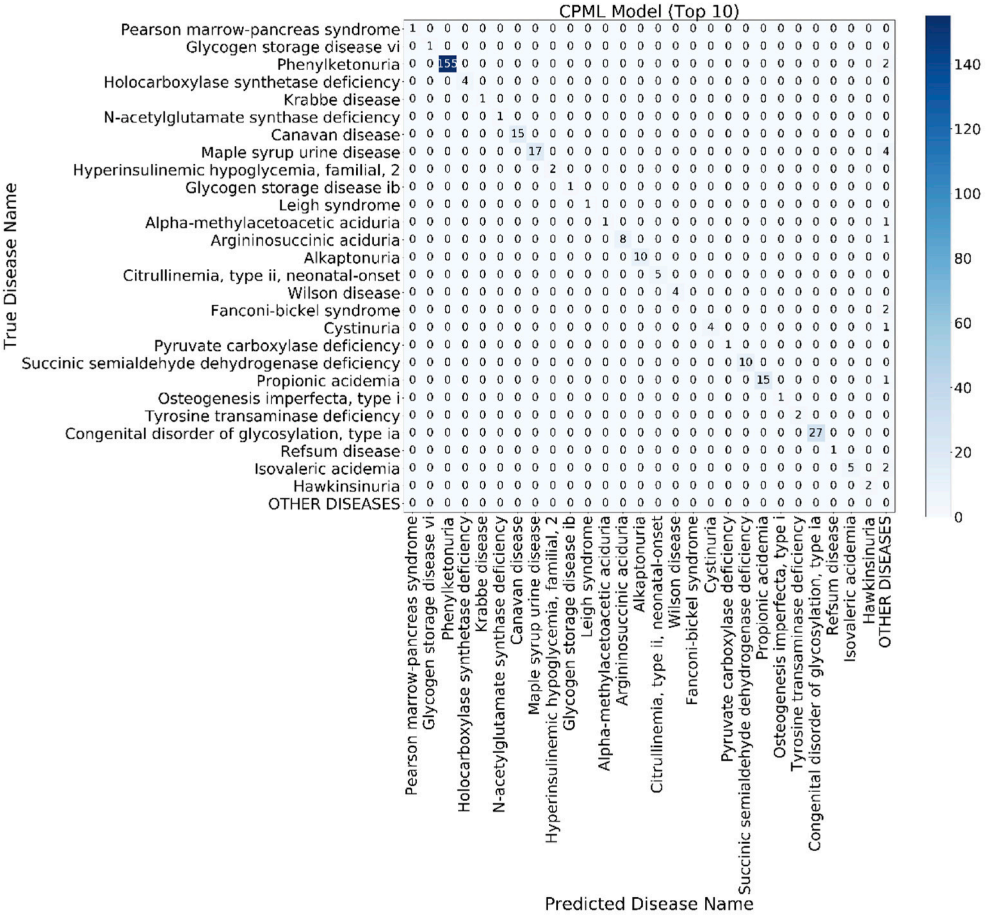


FIGURE 4 | The top 10 candidate rare diseases confusion matrix of the CPML model. The ylab refers to the disease names of the records, while xlab refers to the candidate disease names provided by the diagnostic model.

ranked the most likely candidate rare diseases in the top 10 (**Figure 5**).

Compared with the above results (**Figures 2**, **3**), the result showed that the performance of the classifiers varied in different cases, but where similar to ensemble learning (ensemble learning is a machine learning paradigm where multiple learners are trained to solve the same problem). In contrast to ordinary machine learning approaches that try to learn one hypothesis from training data, ensemble methods try to construct a set of hypotheses and combine them. After using the Bayesian averaging algorithm in the four models to integrate the prediction results of their classifiers, ranking candidate rare diseases with a score, classification results of the four diagnostic models were stable. At the same time, the accuracy and recall rate ranked at the top, varied significantly in the four models built by different data or classifiers. A possible reason for this could be that every patient more or less presents some noise phenotypes and many rare diseases have similar phenotypes, which can interfere with the prediction of the correct rare disease. However, misclassification is significantly reduced when the top 10 is selected as the cutoff value for the predictive outcome, which represents an improvement of the reliability of the model results and is also the designated value we recommend during real application.

DISCUSSION

Rare diseases always have a wide range of complex and diverse phenotypes. However, clinicians always lack knowledge on rare diseases or clinical experiences. Many rare diseases can therefore not be accurately identified on time, and patients are most likely to not receive an accurate diagnosis and subsequent effective treatment. Moreover, due to the heterogeneity of rare diseases, the lack of available clinical diagnostic tests also hinders the timely diagnosis of corresponding diseases. Computer assisted decision support tools have been introduced since the 1960s (Warner, 1989), after which many algorithms were introduced, such as Bayes classifiers (Trace et al., 1990), neural networks (Barnett et al., 1987), rule-based systems (Miller, 1986), and Bayesian networks (Schurink et al., 2005). In this article, we described both the disease similarity method and the machine learning method based diagnostic models for rare disease. We clearly noticed that classifier performance varied in different cases, but similar to ensemble learning, after adopting the Bayesian averaging algorithm in the four models, integrating the prediction results of their classifiers and ranking the candidate rare diseases with score. At the same time, the accuracy and recall rates for all four models built by different data or classifiers,

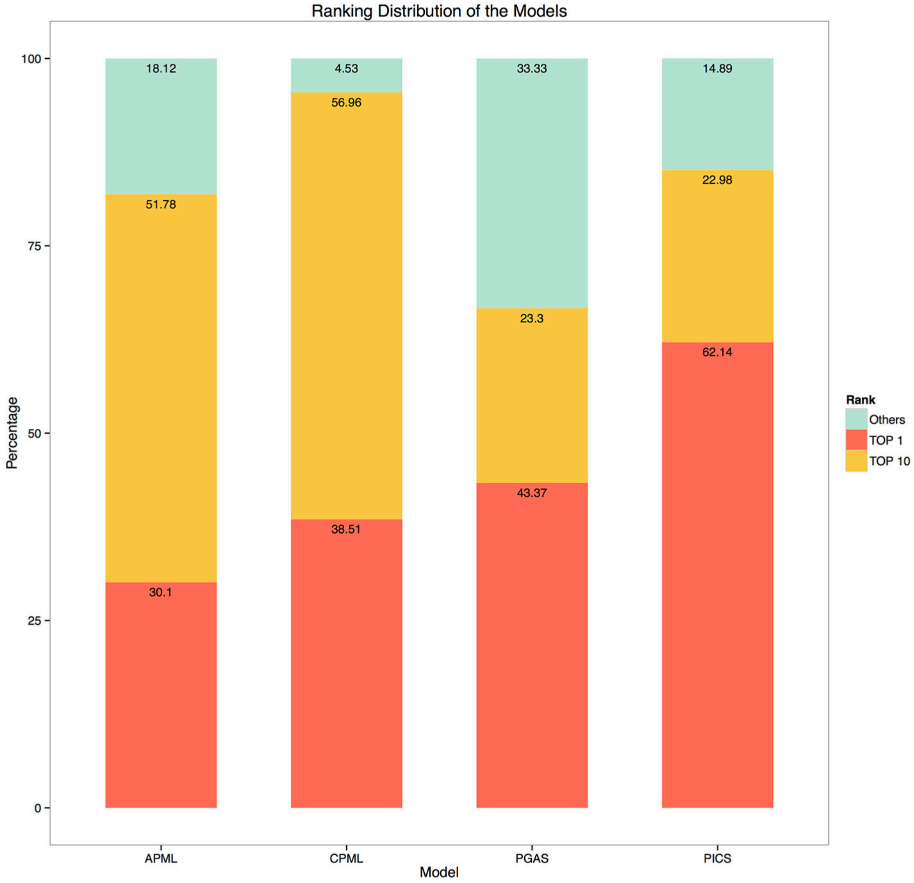


FIGURE 5 | The ranking distribution of the models. The ylab refers to the percentage of disease rankings, while xlab refers to the diagnostic models.

changed significantly when ranked as the top condition, while robustness was ensured when ranked in the top 10 conditions. The reason for this could be that each patient will present some "noise phenotypes," which might interfere with the classification of the model.

Like all the other computer aided diagnosis tools, any rare disease not included in the corresponding model training set cannot be predicted by each diagnostic model contained in the RDAD. In addition, the limited real data sets (EHR/EMR) and diverse patients in this study also restrict the performance of the models. At present, although users are strongly recommended to choose the CPML model in the RDAD system to assist rare disease diagnosis, the RDAD still provides all 4 diagnostic models as alternative to rare disease diagnosis. On the one hand, although the current result show that machine learning models perform better than disease similarity models, PICS performs the best in ranking the top condition (F-1 score 0.73, Precision 0.98 and Recall 0.62). On the other hand, the CPML model performs better than the APML model, but the diagnosis can only be reliable when candidate diseases have corresponding phenotypic annotation in the HPO. For diseases that only have text mined phenotypes, APML will be a better choice; therefore, the four different models can complement each other under different circumstance. It is anticipated that with the accumulation of clinical phenotypes of rare diseases, the performance of our models will improve gradually.

AUTHOR CONTRIBUTIONS

TS supervised the study. JJ and ZA designed and developed the diagnostic models. RW and ZA built up the website. YG and XN verified the validation results. JJ wrote the manuscript. All authors have read and approved the final manuscript.

ACKNOWLEDGMENTS

This work was supported by the China Human Proteomics Project (2014DFB30010,2014DFB30030), the National High Technology Research and Development Program of China (2015AA020108), the National Science Foundation of China (31671377, 31771460, 91629103) and the Shanghai 111 Project (B14019).

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fgene.2018.00587/full#supplementary-material

REFERENCES

Alves, R., Pinol, M., Vilaplana, J., Teixido, I., Cruz, J., et al. (2016). Computer-assisted initial diagnosis of rare diseases. *PeerJ* 4:e2211. doi: 10.7717/peerj.2211

Amberger, J. S., Bocchini, C. A., Schiettecatte, F., Scott, A. F., and Hamosh, A. (2014). OMIM. org: Online Mendelian Inheritance in Man (OMIM®), an online catalog of human genes and genetic disorders. *Nucleic Acids Res.* 43, D789–D798. doi: 10.1093/nar/gku1205

Barnett, G. O., Cimino, J. J., Hupp, J. A., and Hoffer, E. P. (1987). DXplain: an evolving diagnostic decision-support system. *JAMA* 258, 67–74. doi: 10.1001/jama.1987.03400010071030

Dragusin, R., Petcu, P., Lioma, C., Larsen, B., Jorgensen, H. L., et al. (2013). FindZebra: a search engine for rare diseases. *Int. J. Med. Inform.* 82, 528–538. doi: 10.1016/j.ijmedinf.2013.01.005

Ekins, S. (2017). Industrializing rare disease therapy discovery and development. *Nat. Biotechnol.* 35:117. doi: 10.1038/nbt.3787

Firth, H. V., Richards, S. M., Bevan, A. P., Clayton, S., Corpas, M., et al. (2009). DECIPHER: Database of Chromosomal Imbalance and Phenotype in Humans Using Ensembl Resources. *Am. J. Hum. Genet.* 84, 524–533. doi: 10.1016/j.ajhg.2009.03.010

Franco, P. (2013). Orphan drugs: the regulatory environment. *Drug Discov. Today*. 18, 163–172. doi: 10.1016/j.drudis.2012.08.009

Girdea, M., Dumitriu, S., Fiume, M., Bowdin, S., Boycott, K. M., et al. (2013). PhenoTips: patient phenotyping software for clinical and research use. *Hum. Mutat.* 34, 1057–1065. doi: 10.1002/humu.22347

Jia, J., An, Z., Ming, Y., Guo, Y., Li, W., et al. (2018a). eRAM: encyclopedia of rare disease annotations for precision medicine. *Nucleic Acids Res.* 46, D937–D943. doi: 10.1093/nar/gkx1062

Jia, J., An, Z., Ming, Y., Guo, Y., Li, W., et al. (2018b). PedAM: a database for pediatric disease annotation and medicine. *Nucleic Acids Res.* 46, D977–D983. doi: 10.1093/nar/gkx1049

Jia, J., and Shi, T. (2017). Towards efficiency in rare disease research: what is distinctive and important? *Sci. China Life Sci.* 60, 686–691. doi: 10.1007/s11427-017-9099-3

Kohler, S., Schulz, M. H., Krawitz, P., Bauer, S., Dolken, S., et al. (2009). Clinical diagnostics in human genetics with semantic similarity searches in ontologies. *Am. J. Hum. Genet.* 85, 457–464. doi: 10.1016/j.ajhg.2009.09.003

Maiella, S., Olry, A., Hanauer, M., Lanneau, V., Lourghi, H., et al. (2018). Harmonising phenomics information for a better interoperability in the rare disease field. *Eur. J. Med .Genet.* 61, 706–714. doi: 10.1016/j.ejmg.2018.01.013

Miller, R. A. (1986). Quick medical reference (QMR) for diagnostic assistance. *MD. Comput.* 3, 34–48.

Pavan, S., Rommel, K., Marquina, M. E. M., Höhn, S., Lanneau, V., et al. (2017). Clinical practice guidelines for rare diseases: the orphanet database. *PLoS ONE* 12:e0170365. doi: 10.1371/journal.pone.0170365

Pinol, M., Alves, R., Teixid,ó, I., Mateo, J., Solsona, F., et al. (2017). Rare disease discovery: an optimized disease ranking system. *IEEE Trans. Indust. Inform.* 13, 1184–1192. doi: 10.1109/TII.2017.2686380

Qi, Z., Shen, Y., Fu, Q., Li, W., Yang, W., et al. (2017). Whole-exome sequencing identified compound heterozygous variants in MMKS in a Chinese pedigree with Bardet-Biedl syndrome. *Sci. China Life Sci.* 60, 739–745. doi: 10.1007/s11427-017-9085-7

Robinson, P. N., Kohler, S., Bauer, S., Seelow, D., Horn, D., et al. (2008). The human phenotype ontology: a tool for annotating and analyzing human hereditary disease. *Am. J. Hum. Genet.* 83, 610–615. doi: 10.1016/j.ajhg.2008.09.017

Schurink, C., Lucas, P., Hoepelman, I., and Bonten, M. (2005). Computer-assisted decision support for the diagnosis and treatment of infectious diseases in intensive care units. *Lancet Infect. Dis.* 5, 305–312. doi: 10.1016/S1473-3099(05)70115-8

Topel, T., Scheible, D., Trefz, F., and Hofestadt, R. (2010). RAMEDIS: a comprehensive information system for variations and corresponding phenotypes of rare metabolic diseases. *Hum. Mutat.* 31, E1081–1088. doi: 10.1002/humu.21169

Trace, D., Evens, M., Naeymi-Rad, F., and Carmony, L. (1990). Proceedings of the annual symposium on computer application in medical care. *Am. Med. Infor. Assoc.* 635–639.

van Driel, M. A., Bruggeman, J., Vriend, G., Brunner, H. G., and Leunissen, J. A. (2006). A text-mining analysis of the human phenome. *Eur. J. Hum. Genet.* 14, 535–542. doi: 10.1038/sj.ejhg.5201585

Warner, J. H. (1989). Iliad: moving medical decision-making into new frontiers. *Methods Inf. Med.* 28, 370–372. doi: 10.1055/s-0038-1636792

Wu, D., Gong, C., and Su, C. (2017). Genome-wide analysis of differential DNA methylation in Silver-Russell syndrome. *Sci. China Life Sci.* 60, 692–699. doi: 10.1007/s11427-017-9079-7

Xu, R., Li, L., and Wang, Q. (2013). Towards building a disease-phenotype knowledge base: extracting disease-manifestation relationship from literature. *Bioinformatics* 29, 2186–2194. doi: 10.1093/bioinformatics/btt359

Yang, H., Robinson, P. N., and Wang, K. (2015). Phenolyzer: phenotype-based prioritization of candidate genes for human diseases. *Nat. Methods* 12, 841–843. doi: 10.1038/nmeth.3484

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

frontiers in Genetics

ORIGINAL RESEARCH
published: 29 January 2019
doi: 10.3389/fgene.2019.00013

deepDriver: Predicting Cancer Driver Genes Based on Somatic Mutations Using Deep Convolutional Neural Networks

Ping Luo[1], *Yulian Ding*[1], *Xiujuan Lei*[2] *and Fang-Xiang Wu*[1,3,4,5*]

[1] Division of Biomedical Engineering, University of Saskatchewan, Saskatoon, SK, Canada, [2] School of Computer Science, Shaanxi Normal University, Xian, China, [3] School of Mathematics and Statistics, Hainan Normal University, Haikou, China, [4] Department of Mechanical Engineering, University of Saskatchewan, Saskatoon, SK, Canada, [5] Department of Computer Science, University of Saskatchewan, Saskatoon, SK, Canada

With the advances in high-throughput technologies, millions of somatic mutations have been reported in the past decade. Identifying driver genes with oncogenic mutations from these data is a critical and challenging problem. Many computational methods have been proposed to predict driver genes. Among them, machine learning-based methods usually train a classifier with representations that concatenate various types of features extracted from different kinds of data. Although successful, simply concatenating different types of features may not be the best way to fuse these data. We notice that a few types of data characterize the similarities of genes, to better integrate them with other data and improve the accuracy of driver gene prediction, in this study, a deep learning-based method (deepDriver) is proposed by performing convolution on mutation-based features of genes and their neighbors in the similarity networks. The method allows the convolutional neural network to learn information within mutation data and similarity networks simultaneously, which enhances the prediction of driver genes. deepDriver achieves AUC scores of 0.984 and 0.976 on breast cancer and colorectal cancer, which are superior to the competing algorithms. Further evaluations of the top 10 predictions also demonstrate that deepDriver is valuable for predicting new driver genes.

Keywords: deep learning, convolutional neural networks, driver gene prediction, cancer mutations, gene similarity network

Edited by:
Chaoyang Zhang,
University of Southern Mississippi,
United States

Reviewed by:
Luis Rueda,
University of Windsor, Canada
Chao Xu,
University of Oklahoma Health
Sciences Center, United States

****Correspondence:***
Fang-Xiang Wu
faw341@mail.usask.ca

Specialty section:
This article was submitted to
Bioinformatics and Computational
Biology,
a section of the journal
Frontiers in Genetics

Received: *02 August 2018*
Accepted: *11 January 2019*
Published: *29 January 2019*

Citation:
Luo P, Ding Y, Lei X and Wu F-X
(2019) deepDriver: Predicting Cancer
Driver Genes Based on Somatic
Mutations Using Deep Convolutional
Neural Networks. Front. Genet. 10:13.
doi: 10.3389/fgene.2019.00013

1. INTRODUCTION

Cancer is driven by various types of mutations, such as single nucleotide variants (SNVs), insertions or deletions (Indels) and structural variants. Identifying driver genes whose mutations cause cancer could help us decipher the mechanism of cancer, which is beneficial to the development of novel drugs and therapies.

With the advances in next-generation sequencing technologies, massive amounts of cancer genomic data have been published, which elevate the identification of driver genes. Currently, many computational methods have been proposed. Based on their rationale, existing methods can be divided into several types. A typical kind of methods is those based on the mutation frequency. These methods find "significantly mutated genes" (SMG) whose mutation rates are significantly higher than the background mutation rate and judge them as driver genes. For instance, OncodriveCLUST finds positions with mutation rates higher than the background mutation rate and predicts driver genes from clusters generated based on these seed positions (Tamborero et al., 2013). MutsigCV identifies SMGs by building a patient-specific background mutation model with gene expression data and DNA replication time data (Lawrence et al., 2014). However, due to the heterogeneity of tumors, constructing a reliable background mutation model is difficult (Cheng et al., 2015), which limits the performance of frequency-based methods. Another type of methods predicts driver genes by network analysis. For example, DawnRank predicts driver genes by ranking the genes in a gene interaction network (GIN) with PageRank algorithm (Hou and Ma, 2014). SCS uses network control strategy to find driver mutations that can drive the regulation network from the normal state to disease states (Guo et al., 2018). Considering that GINs are downloaded from online databases, such as BioGrid (Chatr-Aryamontri et al., 2017) and HPRD (Keshava Prasad et al., 2008), which contain many false positives, network-based methods need more accurate GIN to improve their prediction accuracy.

As the increasing number of experimentally validated driver genes, researchers start to use machine learning algorithms to predict new driver genes. These methods usually train a classifier with features characterizing the functional impact of mutations. For instance, CHASM trains a random forest classifier with 86 predictive features (Wong et al., 2011). CanDrA trains an SVM with 95 features obtained from 10 functional impact-based algorithms, such as SIFT (Kumar et al., 2009) and CHASM. Since the number of driver genes is much smaller than that of passenger genes, selecting gold-standard driver genes (positive data) and a set of high-quality nonfunctional passenger genes (negative data) is difficult for machine learning-based methods. However, with reasonable downsampling, these methods can also achieve better performance than other types of algorithms. Tokheim et al. propose a random forest algorithm (known as 20/20+) and compare it with seven classical driver gene prediction algorithms [ActiveDriver (Reimand and Bader, 2013), MuSiC (Dees et al., 2012), MutsigCV (Lawrence et al., 2014), OncodriveCLUST (Tamborero et al., 2013), OncodriveFM (Gonzalez-Perez and Lopez-Bigas, 2012), OncodriveFML (Mularoni et al., 2016) and TUSON (Davoli et al., 2013)] in Tokheim et al. (2016). Results show that 20/20+ performs best among the eight algorithms, which demonstrate that machine learning models are able to predict driver genes given the limited known driver-disease associations.

At present, most machine learning-based methods use random forest and SVM as the classifier. To improve the prediction accuracy, various kinds of features extracted from different types of data are used to train the classifier. Despite the increase of the dimensionality, simply concatenating all these features may not be the best approach to integrate different types of data. Considering that several types of data can be used to characterize the similarities of genes, if we construct similarity networks with these data and combine them with other predictive features, the prediction accuracy of the algorithms should be improved compared to that obtained from a simple feature concatenation. Thus, in this study, a deep learning-based method is proposed to predict driver genes by combining similarity networks with features that characterize the functional impact of mutations (deepDriver). Specifically, candidate driver genes are predicted by a convolutional neural network (CNN) trained with mutation-based feature matrix constructed based on the topological structure of a similarity network. The algorithm leverages the similarity of gene expression patterns and the functional impact of mutations simultaneously, which can better fuse these two types of data and improve the prediction accuracy. To our knowledge, this is the first time that CNN is combined with similarity network to predict driver genes.

In the rest of the paper, section 2 describes the materials and methods used in the study. Section 3 analyzes the results of the evaluation. Section 4 draws some conclusions.

2. MATERIALS AND METHODS

2.1. General Model

CNN is successful in many areas, such as image classification and speech recognition. The key component of a CNN is the convolutional (CONV) layer, which helps the model to learn local and global structures from the input data. In an image classification problem, these structures include edges, curves, corners, etc. While in a driver gene prediction problem, traditional input data contain distinct features that characterize different properties of genes, which cannot be directly applied to CNN.

We notice that pixels in a small region share the same filters because they have similar grayscale. In a gene similarity network (GSN), genes and their neighbors also have similar properties. If we reconstruct the traditional input data with GSN so that features of similar genes are close to each other, CNN can then be applied to these reconstructed data. Instead of edges and curves learned from the images, topological structures of the similarity networks are learned by CNN with this strategy. In addition, the strategy allows CNN to learn the similarities of genes and the properties of the original input data simultaneously, which can improve the accuracy of driver gene prediction.

Figure 1 depicts a schematic example of a 1-dimensional CNN, which is used in our study. The model consists of five kinds of layers: Input layer, CONV layers, pooling layers, Fully-Connected (FC) layers, and Output layer. Given a feature matrix $\phi_i \in R^{2k \times n_f}$ constructed by the feature vectors of g_i and its k neighbors where n_f is the dimension of the feature vectors of g_i, the output of a CONV layer corresponds to the input ϕ_i and the filter w_j is calculated as follows

$$A(i,j) = f(w_j \phi_i + b_j) \tag{1}$$

where b_j denotes the bias corresponding to w_j, f is an activation function which is ReLU in this study. $w_j \phi_i$ is still the dot product of w_j and ϕ_i except that the calculation is restricted to be local spatially. Each CONV layer is followed by a pooling layer, and the CONV-POOL pattern is repeated for several times. The final structure of the model used for driver gene prediction is

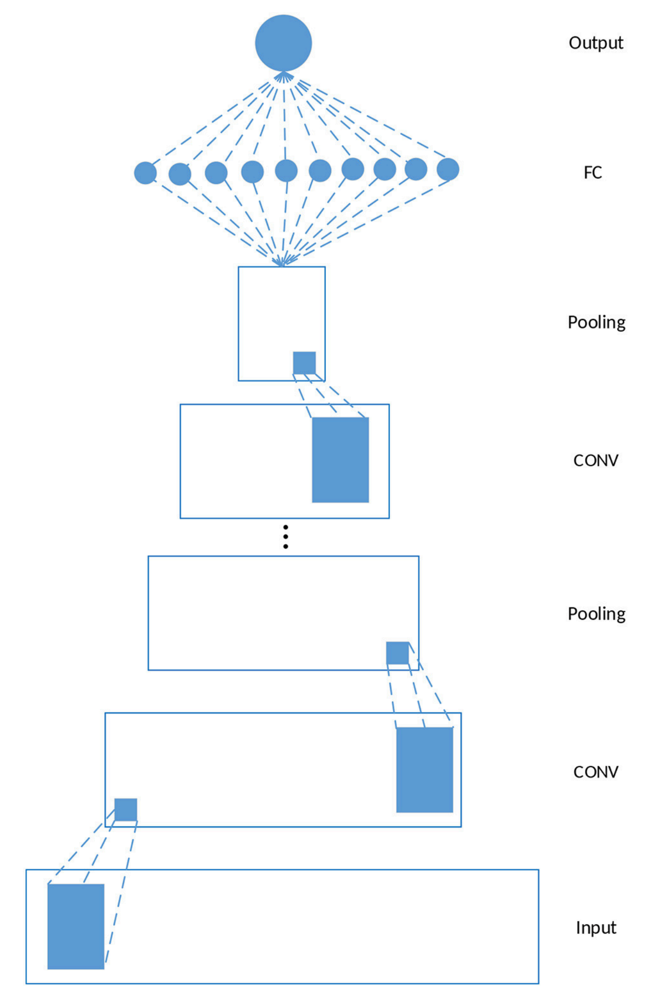


FIGURE 1 | Schematic 1-D CNN. In this study, each CONV layer is followed by a pooling layer and the CONV-POOL pattern is repeated for several times. The final structure of the model is determined by grid search.

determined by grid search, and the results are discussed in section 3.2. The construction of ϕ_i is discussed in the next section.

2.2. Network-Based Convolution

The convolution is performed by combining mutation-based features with gene similarity networks. Many approaches can be used to calculate the similarities of genes. In this study, to characterize the relationships between genes in the disease states, Pearson correlation coefficient (PCC) defined by the following equation is used to calculate the similarities.

$$r(g_i, g_j) = \frac{\sum_{q=1}^{v}(e_{iq} - \bar{e}_i)(e_{jq} - \bar{e}_j)}{\sqrt{\sum_{q=1}^{v}(e_{iq} - \bar{e}_i)^2}\sqrt{\sum_{q=1}^{v}(e_{jq} - \bar{e}_j)^2}} \tag{2}$$

where $\mathbf{e}_i = (e_{i1}, e_{i2}, \ldots, e_{iv})$ denotes the expression values of g_i in v tumor samples, and $\bar{\mathbf{e}}_i$ is the mean of $\mathbf{e}_i$. An undirected network N is constructed by k-nearest neighbors (kNN) algorithm (Cover and Hart, 1967) in which every gene is connected to genes that have the k largest PCC scores with itself.

After obtaining N, the construction of ϕ_i used in the convolution is depicted by **Figure 2**. Assuming we have obtained a feature vector x_i for each gene g_i, and $g_{s1}, g_{s2}, \ldots, g_{sk}$ are the k nearest neighbors of g_i in N, where $pcc(g_i, g_{s1}) > pcc(g_i, g_{s2}) > \cdots > pcc(g_i, g_{sk})$. Feature matrix $\phi_i \in R^{2k \times n_f}$ is built as depicted by the figure. In ϕ_i, features of similar genes are close to each other so that they can share the same filters in the CONV layer.

2.3. Mutation-Based Features

For each gene of a specific disease, 12 features are extracted from the mutation datasets. **Table 1** lists the names and descriptions of these features. Among them, the first eight ones measure the fraction of a specific type of mutation among all the mutations. The tenth and eleventh feature measure the rate of missense mutations and non-silent mutations to silent mutations, respectively. The last two features measure the positional clustering of different types of mutations and are calculated as follows

$$E_i = \frac{-\sum_i p_j \log_2 p_j}{\log_2 m} \tag{3}$$

For the normalized missense entropy, m is the total number of missense mutations of g_i, and $p_j = \kappa_j/m$ where κ_j is the number of missense mutations in the j-th codon. For the normalized mutation entropy, m is the total number of all types of mutations of g_i. Different mutations are binned together based on their types, except for that missense mutations are binned based on their codon positions, different silent mutations are divided into their own bins. Inactivating mutations (nonsense, translation start site, nonstop, splice site) are grouped into a single bin.

These 12 features have been used in many machining learning-based methods (Vogelstein et al., 2013; Tokheim et al., 2016). To demonstrate the superiority of our model, we did not use any other features proposed by specific methods. In addition, during the implementation of the competing methods (SVM, 20/20+), only these 12 features are used to train their models.

2.4. Data Sources

In this study, deepDriver was evaluated on three types of cancer: breast invasive carcinoma (BRCA), colon adenocarcinoma (COAD) and lung adenocarcinoma (LUAD). The mutation data and gene expression data of these three diseases were downloaded from the NCI Genomic Data Commons (GDC) (Grossman et al., 2016). For the mutation data, quality control was applied by filtering out hypermutated samples ($> 1,000$ intragenic somatic variants) (Vogelstein et al., 2013). In total, 228,046, 168,746, and 287,667 somatic variants were obtained for BRCA, COAD, and LUAD, respectively. For gene expression, datasets of 1,102 BRCA, 478 COAD and 551 LUAD primary tumor samples measured by RNA-Seq were downloaded. We chose the data normalized by FPKM and converted the values to TPM by the method proposed in Pachter (2011). Three steps were then performed to remove the genes that are barely expressed in tumor samples. First, TPM values <1 were considered unreliable and replaced

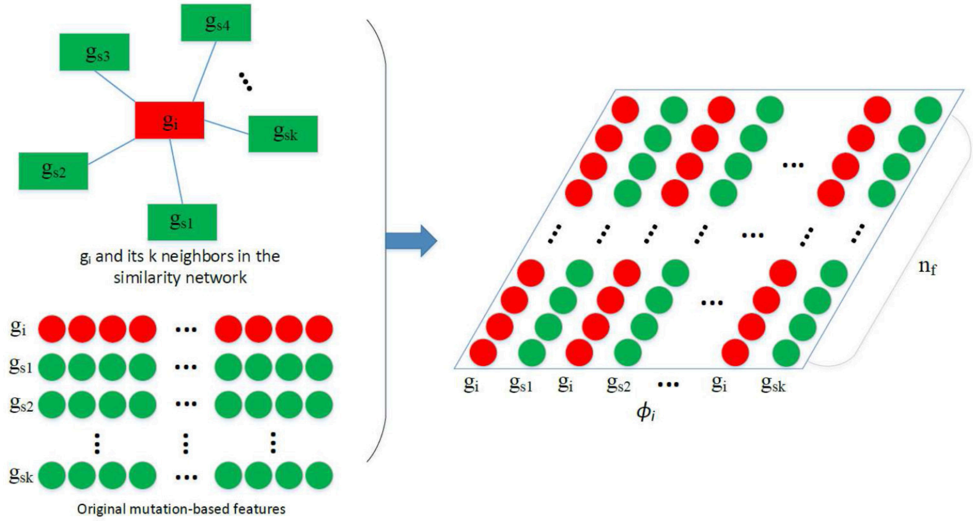

g_s4
g_s3
g_i
g_sk
g_s2
g_s1
g_i and its k neighbors in the similarity network
g_i
g_s1
g_s2
g_sk
Original mutation-based features
n_f
g_i g_s1 g_i g_s2 ••• g_i g_sk
ϕ_i

FIGURE 2 | Construction of ϕ_i. Given the feature vectors of g_i and its k nearest neighbors $g_{s1}, g_{s2}, \ldots, g_{sk}$, a feature matrix ϕ_i is constructed by arranging the $2k$ vectors into a $2k \times n_f$ matrix, which is then used in the convolution.

TABLE 1 | Twelve features extracted from mutation data.

No.	Name	Description
1	Silent fraction	Fraction of silent mutations
2	Nonsense fraction	Fraction of nonsense mutations
3	Splice site fraction	Fraction of splice site mutations
4	Missense fraction	Fraction of missense mutations
5	Recurrent missense fraction	Fraction of recurrent missense mutations
6	Frameshift indel fraction	Fraction of frameshift indel mutations
7	Inframe indel fraction	Fraction of inframe indel mutations
8	Lost start and stop fraction	Fraction of lost start and stop mutations
9	Missense to silent	Ratio of missense to silent mutations
10	Non-silent to silent	Ratio of non-silent to silent mutations
11	Normalized missense position entropy	See section 2.3
12	Normalized mutation entropy	See section 2.3

by 0. Second, $\log_2(\text{TPM} + 1)$ was applied to all TPM values. Third, genes expressed in $<$ 10% of all tumor samples were removed.

Gene ids were standardized to the gene names provided by HUGO Gene Nomenclature Committee (downloaded Aug 1, 2018) (Yates et al., 2016). Only genes that have both mutation and expression data are kept. Finally, 13,777 genes for BRCA, 11,282 genes for COAD, and 13,731 genes for LUAD passed the quality control.

The driver genes were collected from two sources—the Cancer Gene Census category (CGC) (Forbes et al., 2016) and the genes published in Bailey et al. (2018). Genes in CGC were divided into two tiers, and we used genes in Tier 1 as driver genes because strong evidence has proved their oncogenic role in cancer genesis. It is of note that both oncogene and tumor suppressor gene (TSG) are regarded as driver gene in this study. In total, 37 driver genes for BRCA, 42 driver genes for COAD and 12 driver genes for LUAD were collected from CGC. The Bailey et al.'s dataset (Bailey et al., 2018) contains 299 driver genes associated with 33 types of cancer. In total, 29 driver genes for BRCA, 20 driver genes for COAD and 20 driver genes for LUAD were collected.

To validate the performance of the algorithm, the structure of the model was first determined by the grid search using the driver genes of BRCA and COAD collected from CGC. Then, the optimal model was directly applied to LUAD without fine-tuning the hyperparameters. Similarly, when the model was trained with the driver genes published in Bailey et al. (2018), the optimal hyperparameters were used without fine-tuning.

2.5. Evaluation Metrics

The algorithm was evaluated in two steps. In the first step, deepDriver was compared with 20/20+ and SVM in terms of the AUC (area under the receiver operating characteristic (ROC) curve) scores obtained from 10-fold cross-validation. ROC curve plots the false positive rate (FPR) against the true positive rate (TPR) at different thresholds. FPR and TPR are defined as follows

$$\begin{aligned} FPR &= \frac{FP}{FP + TN} \\ TPR &= \frac{TP}{TP + FN} \end{aligned} \tag{4}$$

where TP, FP, TN, and FN are the numbers of true positives, false positives, true negatives, and false negatives, respectively. In this study, a true positive is a driver gene predicted as a driver gene, a false positive is a passenger gene predicted as driver gene, a true negative is a passenger gene predicted as a passenger gene, and a false negative is a driver gene predicted as a passenger gene. Algorithm with the highest AUC score performs the best.

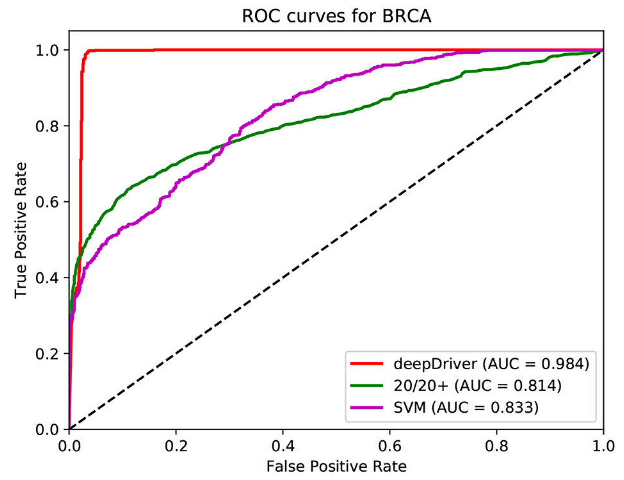


FIGURE 3 | ROC curves of the three algorithms obtained on the dataset of BRCA. The red, green, and magenta lines depict the ROC curves of deepDriver, 20/20+ and SVM, respectively. The AUC value of deepDriver is 0.984, which is at least 15.1% higher than that of the other two algorithms.

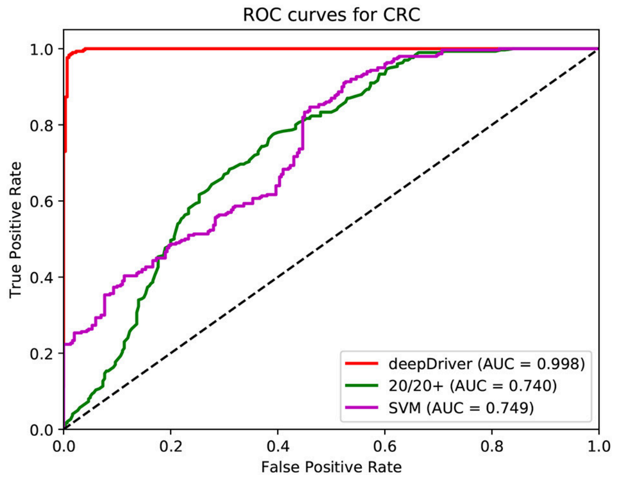


FIGURE 5 | ROC curves of the three algorithms obtained on the dataset of LUAD. The red, green, and magenta lines depict the ROC curves of deepDriver, 20/20+ and SVM, respectively. The AUC value of deepDriver is 0.998, which is at least 24.9% higher than that of the other two algorithms.

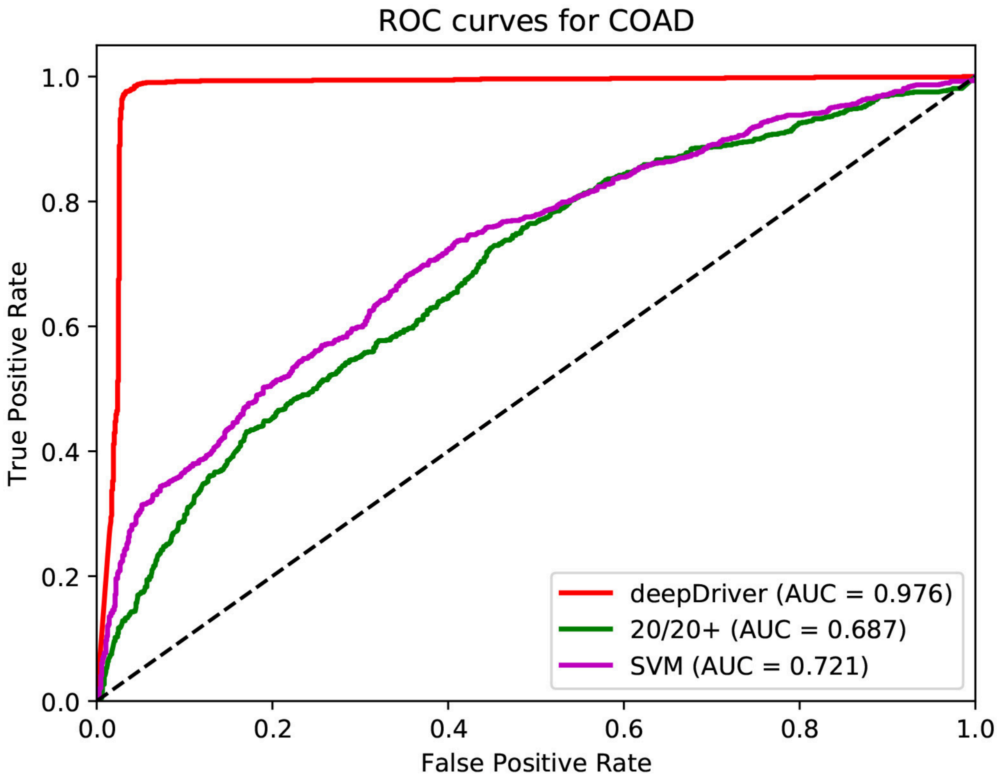


FIGURE 4 | ROC curves of the three algorithms obtained on the dataset of COAD. The red, green, and magenta lines depict the ROC curves of deepDriver, 20/20+ and SVM, respectively. The AUC value of deepDriver is 0.976, which is at least 25.5% higher than that of the other two algorithms.

Since the number of passenger genes is much larger than that of the driver genes, a method is needed to solve the imbalanced issue. Currently, two types of methods can be used to solve the imbalanced problem: data level methods and classifier level methods (Buda et al., 2018). In this study, a data level method, downsampling, was used to reduce the size of the passenger genes. Specifically, a subset of passenger genes was randomly selected from all the passengers so that the numbers of positive samples (driver genes) and negative samples (passenger genes) are equal. This approach was run for five times which generated five sets of data. During the cross-validation, for each set of data, all the positive and negative samples were randomly split into ten groups, and the CNN model was validated for ten rounds. In each round, one group of samples were used as the testing data while the rest nine groups of samples were used as the training data.

Additionally, since passenger genes are barely reported in existing literature, in this study, genes that have not been reported as cancer driver genes (unknown genes) were regarded as passenger genes. This strategy was used because of the following two reasons. First, the numbers of the selected passenger genes and the undiscovered driver genes are both much less than that of the unknown genes. Potential driver genes only have a small change to be selected as passenger genes (Davoli et al., 2013). Second, the final results were obtained by taking the average predictions of the five sets of data. This bagging strategy would improve the stability and accuracy of the results and reduce the impact of a potential driver gene selected as a passenger gene. Finally, the 10-fold cross-validation was run for five times for each dataset to reduce the influence of random shuffling, and the average AUC score was used to evaluate the performance of the algorithms.

In the second step, all the unknown genes were ranked by their probabilities of being driver genes, and the top 10 predictions were searched from the existing literature to check whether our predictions are in concert with existing studies. We also ranked the unknown genes by SVM, 20/20+ and OncodriveCLUST and compared their results with those of deepDriver in terms of the number of genes having been analyzed in existing literature.

2.6. Implementation

The algorithm was implemented using Keras (Chollet, 2015) with TensorFlow (Abadi et al., 2015) as the backend engine. We have tested the program on both CPU and GPU versions of TensorFlow and the model can be efficiently trained with or without the help of GPU. A reference implementation is available at GitHub.

3. RESULTS

3.1. Hyperparameters

In this study, the architecture of CNN is determined by the following hyperparameters.

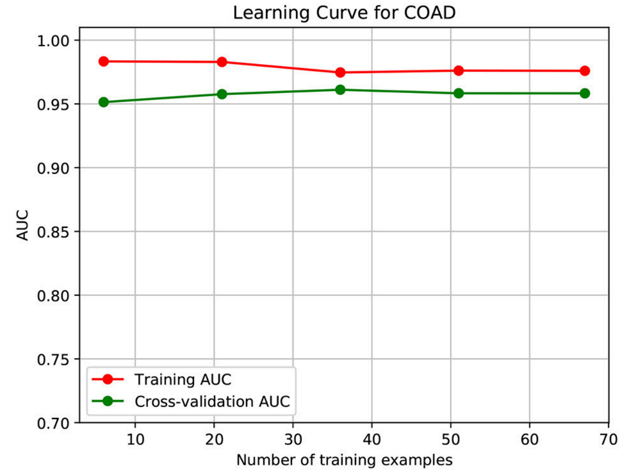


FIGURE 7 | Learning curve for COAD.

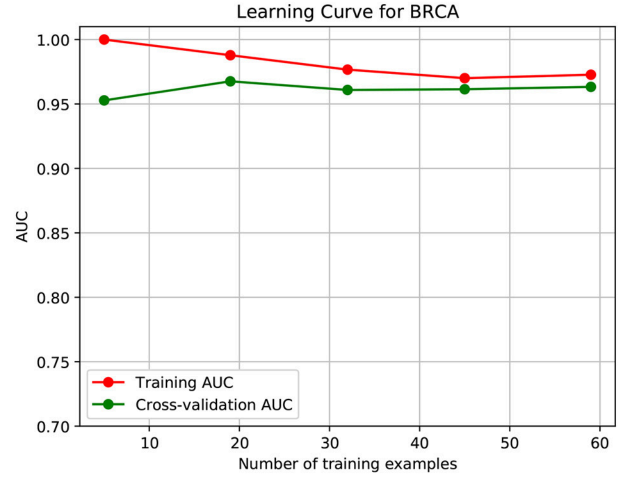


FIGURE 6 | Learning curve for BRCA.

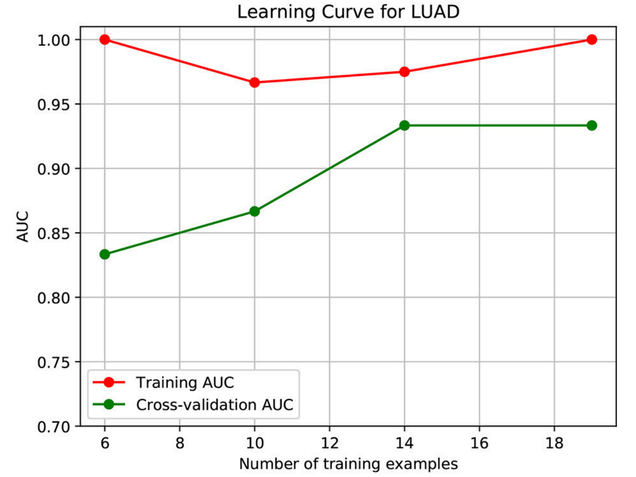


FIGURE 8 | Learning curve for LUAD.

1. The number of the CONV layers (*ncl*)
2. The number of the FC layers (*nfl*)
3. The number of the nodes in the CONV layers (*ncn*)
4. The number of the nodes in the FC layers (*nfn*)

These hyperparameters were determined by grid search, with *ncl* searched from {1, 2, 3, 4}, *nfl* searched from {1, 2, 3}, *ncn* searched from {12, 24, 48} and *nfn* searched from {24, 48, 96}. The optimal values of *ncl*, *nfl*, *ncn*, and *nfn* are 2, 1, 24, and 48, respectively. In addition, zero padding was used in the CONV layers except the first one. The size of the filters, the window size of the pooling layers and the stride sizes used in the CONV layers and the pooling layers were all empirically set to 2.

The number of neighbors used by kNN algorithms was also determined by grid search. We searched k from {3, 5, 7, 9, 11, 13, 15}, and finally, $k = 9$ and $k = 7$ were chosen for BRCA and COAD, respectively. In fact, the AUC scores were all above 0.950 when $7 \leq k \leq 15$. Based on our previous study, $k = 7$ is enough to generate high-quality similarity networks (Luo et al., 2017). Thus, $k = 7$ was used when the dataset of LUAD was analyzed by our deepDriver. Meanwhile, for other types of cancer not discussed in this study, $k = 7$ is also recommended when the similarity network is constructed.

For 20/20+, a random forest of 200 trees was used based on the suggestions of Tokheim et al. (2016). For SVM, the model was implemented with a linear kernel and RBF kernel. The penalty parameter C was searched from {0.1, 0.01, 0.001, 1, 10, 100, 1,000}, and γ was searched from {1/12, 0.001, 0.0001, 0.00001}. Finally, for BRCA and COAD, SVM performed the best with an RBF kernel, when $C = 1, \gamma = 0.0001$; for LUAD, SVM performed the best with an RBF kernel, when $C = 1,000, \gamma = 0.00001$.

3.2. Cross-Validation

Figures 3–5 show the results of the ROC curves and the corresponding AUC scores of deepDriver, 20/20+ and SVM on BRCA, COAD and LUAD, respectively. According to the figures, deepDriver achieved AUC scores of 0.984, 0.976, and 0.998 on BRCA, COAD, and LUAD, respectively, which were at least 15.1% higher than those of the two competing algorithms, especially for COAD and LUAD where the AUC scores of the competing algorithms were <0.750.

To further demonstrate that the model was not overfitted, the learning curves were plotted using the datasets of the three types of cancer. For each type of cancer, 80% of the total samples were used as training data while the rest 20% samples were left to test the performance of the model. **Figures 6–8** show the results of the learning curves. The AUC scores obtained from the testing set improved with the increase of the number of the training samples, which demonstrates that the model is not overfitted. In the meantime, the AUC scores obtained with a small amount of samples also demonstrate that the model is able to produce meaningful results even if the number of the known driver genes is <10.

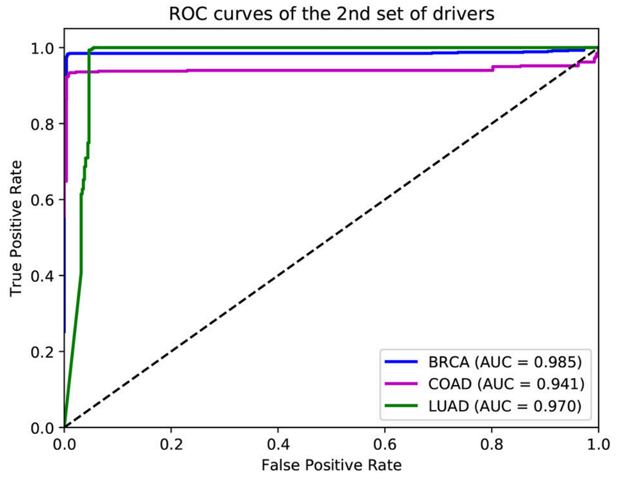


FIGURE 9 | ROC curves of deepDriver obtained from the second sets of driver genes.

TABLE 2 | Top 10 predictions of deepDriver.

Gene names	References
BRCA	
PTEN	Kechagioglou et al., 2014
HCFC1	Gonzalez-Perez et al., 2013; Rubio-Perez et al., 2015
UTRN	Cornen et al., 2014
ZNF517	
STAG2	Gonzalez-Perez et al., 2013; Rubio-Perez et al., 2015
ZFP36L1	Loh et al., 2017
ZNF91	
VPS13C	
DST	
FBXW7	Cao et al., 2016
COAD	
AMER1	
SOX9	Prévostel and Blache, 2017
NRAS	Meriggi et al., 2014
MTOR	Wang and Zhang, 2014
ATM	AlDubayan et al., 2018
ADAMTSL3	
ELMO1	Zheng et al., 2017
TG	
LAMA3	
KMT2A	
LUAD	
XIST	Wang et al., 2017
MALAT1	Li et al., 2018
STK11	Pécuchet et al., 2017
USH1C	
HSP90AB2P	
BNIP3P1	
EEF1A1P9	
UBE2MP1	
SMAD4	Haeger et al., 2016
HERC2P3	

In addition to the driver genes collected from CGC, our deepDriver was also validated using the driver genes published in Bailey et al. (2018). As discussed in section 2.4, the optimal hyperparameters obtained from the first set of drivers were directly used to evaluate the model. **Figure 9** depicts the resulted ROC curves. Our deepDriver obtained AUC scores of 0.985, 0.941, and 0.970 on BRCA, COAD, and LUAD, respectively.

3.3. *De novo* Study

To further evaluate the performance of deepDriver, the unknown genes were ranked by their probabilities of being driver genes predicted by the model. Similar to the cross-validation, 5 sets of data were used to train the model and the unknown genes were ranked by the average probabilities. Meanwhile, we also ranked the unknown genes using the three competing algorithms and compared their results with those of deepDriver in terms of the number of genes that have been studied as drivers in existing literature.

Table 2 shows the top 10 predicted driver genes of deepDriver. Six out of the 10 genes have been studied in existing literature or databases as potential driver genes of BRCA. The ninth gene "DST" was found to have the potential to drive ductal carcinoma *in situ* to breast cancer (Lee et al., 2012). Five out of the 10 genes have been studied as driver genes of COAD in the existing literature. Meanwhile, among the rest 5 genes, "AMER1" and "ADAMTSL3" were found to be frequently mutated in COAD (Koo et al., 2007; Sanz-Pamplona et al., 2015). "LAMA3" were predicted as biomarkers which could be used to diagnose COAD in the early stage (Choi et al., 2015). "KMT2A" belongs to the KMT2 family which is related to COAD (Rao and Dou, 2015). Four out of 10 genes have been studied as driver genes of LUAD. The tenth gene "HERC2P3" contains a microsatellite locus

TABLE 3 | Top 10 predictions of 20/20+.

Gene names	References
BRCA	
KMT2C	Gala et al., 2018
PTEN	Kechagioglou et al., 2014
ANKRD12	
NF1	Uusitalo et al., 2017
ANKHD1-EIF4EBP3	
ARID4B	
MCM7	
MYO6	
MLLT4	Gonzalez-Perez et al., 2013
CEP128	
COAD	
ATM	AlDubayan et al., 2018
SOX9	Prévostel and Blache, 2017
LAMA3	
ADAMTSL3	
ELMO1	Zheng et al., 2017
OLFM1	
BRINP1	
ACVR1B	
CNOT1	
PCDH7	
LUAD	
LRRIQ1	
HECTD4	
EPB41L3	Kikuchi et al., 2005
NF1	Redig et al., 2016
CEP350	
PRKDC	
APC	
MYH9	
POSTN	
FN1	

TABLE 4 | Top 10 predictions of SVM.

Gene names	References
BRCA	
VPS13C	
UTRN	Cornen et al., 2014
HCFC1	Gonzalez-Perez et al., 2013; Rubio-Perez et al., 2015
MLLT4	Gonzalez-Perez et al., 2013
ZNF91	
STAG2	Gonzalez-Perez et al., 2013; Rubio-Perez et al., 2015
FBXW7	Cao et al., 2016
MALAT1	
NRK	
BAZ2B	
COAD	
ATM	AlDubayan et al., 2018
NRAS	Meriggi et al., 2014
MTOR	Wang and Zhang, 2014
SOX9	Prévostel and Blache, 2017
ADAMTSL3	
ELMO1	Zheng et al., 2017
AMER1	
KMT2B	
FBN2	
KMT2A	
LUAD	
XIST	Wang et al., 2017
MALAT1	Li et al., 2018
USH1C	
SNRPN	
STK11	Pécuchet et al., 2017
SMAD4	Haeger et al., 2016
POLA1	
MAGEE1	
BRAF	
CTNNB1	

TABLE 5 | Top 10 predictions of OncodriveCLUST.

Gene names	References
BRCA	
ACTN4	Honda, 2015
AFF2	
ATP2B3	
AVPR1B	
CASR	
CMYA5	
DIS3L	
EPB41L2	
FBXW8	
KCND3	
COAD	
AKAP12	He et al., 2018
C3orf20	
COL1A2	Yu et al., 2018
DOK1	Friedrich et al., 2016
FNDC1	
MSRB3	
NCOA2	Yu et al., 2016
NPHS1	
NRAP	
PCDHB13	

that can precisely discriminate LUAD samples and non-tumor samples (Velmurugan et al., 2017). As for three competing algorithms, **Tables 3–5** show their prediction results. In summary, deepDriver performed better than the three competing algorithms in predicting new cancer drivers. Its prediction results were in concert with existing studies which further reveal the value of deepDriver in predicting cancer driver genes.

4. CONCLUSION

In this study, we proposed an algorithm to predict cancer driver genes with CNN. The method combined CNN with similarity networks so that the functional impact of mutations and similarities of gene expression can be learned simultaneously, which improve the accuracy of driver gene prediction. Experiments performed on BRCA, COAD, and LUAD then showed that deepDriver was superior to the competing algorithms in terms of both cross-validation and *de novo* prediction.

In the future, similarity networks calculated by different strategies and predictive features extracted by other algorithms can both be used to improve the prediction accuracy. Meanwhile, the algorithm can be applied to the pancancer dataset to predict generic cancer driver genes. Since the total number of cancer driver genes is much higher than that of a specific type of cancer, candidate driver genes can also be further classified into TSG and oncogene on the pancancer dataset.

AUTHOR CONTRIBUTIONS

F-XW conceived this study. F-XW, PL, YD, and XL discussed about the methods. PL implemented the algorithm, designed and performed the experiments. PL and F-XW wrote the manuscript. All authors read and approved the final manuscript.

FUNDING

This work is supported in part by Natural Science and Engineering Research Council of Canada (NSERC) and China Scholarship Council (CSC).

REFERENCES

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., et al. (2015). *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Available online at: https://www.tensorflow.org

AlDubayan, S. H., Giannakis, M., Moore, N. D., Han, G. C., Reardon, B., Hamada, T., et al. (2018). Inherited dna-repair defects in colorectal cancer. *Am. J. Hum. Genet.* 102, 401–414. doi: 10.1016/j.ajhg.2018.01.018

Bailey, M. H., Tokheim, C., Porta-Pardo, E., Sengupta, S., Bertrand, D., Weerasinghe, A., et al. (2018). Comprehensive characterization of cancer driver genes and mutations. *Cell* 173, 371–385. doi: 10.1016/j.cell.2018.02.060

Buda, M., Maki, A., and Mazurowski, M. A. (2018). A systematic study of the class imbalance problem in convolutional neural networks. *Neural Netw.* 106, 249–259. doi: 10.1016/j.neunet.2018.07.011

Cao, J., Ge, M.-H., and Ling, Z.-Q. (2016). Fbxw7 tumor suppressor: a vital regulator contributes to human tumorigenesis. *Medicine* 95:e2496. doi: 10.1097/MD.0000000000002496

Chatr-Aryamontri, A., Oughtred, R., Boucher, L., Rust, J., Chang, C., Kolas, N. K., et al. (2017). The biogrid interaction database: 2017 update. *Nucleic Acids Res.* 45, D369–D379. doi: 10.1093/nar/gkw1102

Cheng, F., Zhao, J., and Zhao, Z. (2015). Advances in computational approaches for prioritizing driver mutations and significantly mutated genes in cancer genomes. *Brief. Bioinformatics* 17, 642–656. doi: 10.1093/bib/bbv068

Choi, M. R., An, C. H., Yoo, N. J., and Lee, S. H. (2015). Laminin gene lamb 4 is somatically mutated and expressionally altered in gastric and colorectal cancers. *Apmis* 123, 65–71. doi: 10.1111/apm.12309

Chollet, F. (2015). *Keras*. Available online at: https://keras.io

Cornen, S., Guille, A., Adélaïde, J., Addou-Klouche, L., Finetti, P., Saade, M.-R., et al. (2014). Candidate luminal b breast cancer genes identified by genome, gene expression and dna methylation profiling. *PLoS ONE* 9:e81843. doi: 10.1371/journal.pone.0081843

Cover, T., and Hart, P. (1967). Nearest neighbor pattern classification. *IEEE Trans. Inf. Theory* 13, 21–27. doi: 10.1109/TIT.1967.1053964

Davoli, T., Xu, A. W., Mengwasser, K. E., Sack, L. M., Yoon, J. C., Park, P. J., et al. (2013). Cumulative haploinsufficiency and triplosensitivity drive aneuploidy patterns and shape the cancer genome. *Cell* 155, 948–962. doi: 10.1016/j.cell.2013.10.011

Dees, N. D., Zhang, Q., Kandoth, C., Wendl, M. C., Schierding, W., Koboldt, D. C., et al. (2012). Music: identifying mutational significance in cancer genomes. *Genome Res.* 22, 1589–1598. doi: 10.1101/gr.134635.111

Forbes, S. A., Beare, D., Boutselakis, H., Bamford, S., Bindal, N., Tate, J., et al. (2016). Cosmic: somatic cancer genetics at high-resolution. *Nucleic Acids Res.* 45, D777–D783. doi: 10.1093/nar/gkw1121

Friedrich, T., Söhn, M., Gutting, T., Janssen, K.-P., Behrens, H.-M., Röcken, C., et al. (2016). Subcellular compartmentalization of docking protein-1 contributes to progression in colorectal cancer. *EBioMedicine* 8, 159–172. doi: 10.1016/j.ebiom.2016.05.003

Gala, K., Li, Q., Sinha, A., Razavi, P., Dorso, M., Sanchez-Vega, F., et al. (2018). Kmt2c mediates the estrogen dependence of breast cancer through regulation of erα enhancer function. *Oncogene* 37, 4692–4710. doi: 10.1038/s41388-018-0273-5

Gonzalez-Perez, A., and Lopez-Bigas, N. (2012). Functional impact bias reveals cancer drivers. *Nucleic Acids Res.* 40, e169–e169. doi: 10.1093/nar/gks743

Gonzalez-Perez, A., Perez-Llamas, C., Deu-Pons, J., Tamborero, D., Schroeder, M. P., Jene-Sanz, A., et al. (2013). Intogen-mutations identifies cancer drivers across tumor types. *Nat. Methods* 10, 1081. doi: 10.1038/nmeth.2642

Grossman, R. L., Heath, A. P., Ferretti, V., Varmus, H. E., Lowy, D. R., Kibbe, W. A., et al. (2016). Toward a shared vision for cancer genomic data. *New Engl. J. Med.* 375, 1109–1112. doi: 10.1056/NEJMp1607591

Guo, W.-F., Zhang, S.-W., Liu, L.-L., Liu, F., Shi, Q.-Q., Zhang, L., et al. (2018). Discovering personalized driver mutation profiles of single samples in cancer by network control strategy. *Bioinformatics* 34, 1893–1903. doi: 10.1093/bioinformatics/bty006

Haeger, S. M., Thompson, J. J., Kalra, S., Cleaver, T. G., Merrick, D., Wang, X.-J., et al. (2016). Smad4 loss promotes lung cancer formation but increases sensitivity to dna topoisomerase inhibitors. *Oncogene* 35:577. doi: 10.1038/onc.2015.112

He, P., Li, K., Li, S.-B., Hu, T.-T., Guan, M., Sun, F.-Y., et al. (2018). Upregulation of akap12 with hdac3 depletion suppresses the progression and migration of colorectal cancer. *Int. J. Oncol.* 52, 1305–1316. doi: 10.3892/ijo.2018.4284

Honda, K. (2015). The biological role of actinin-4 (actn4) in malignant phenotypes of cancer. *Cell Biosci.* 5:41. doi: 10.1186/s13578-015-0031-0

Hou, J. P., and Ma, J. (2014). Dawnrank: discovering personalized driver genes in cancer. *Genome Med.* 6:56. doi: 10.1186/s13073-014-0056-8

Kechagioglou, P., Papi, R. M., Provatopoulou, X., Kalogera, E., Papadimitriou, E., Grigoropoulos, P., et al. (2014). Tumor suppressor pten in breast cancer: heterozygosity, mutations and protein expression. *Anticancer Res.* 34, 1387–1400.

Keshava Prasad, T., Goel, R., Kandasamy, K., Keerthikumar, S., Kumar, S., Mathivanan, S., et al. (2008). Human protein reference database 2009 update. *Nucleic Acids Res.* 37(Suppl. 1):D767–D772. doi: 10.1093/nar/gkn892

Kikuchi, S., Yamada, D., Fukami, T., Masuda, M., Sakurai-Yageta, M., Williams, Y. N., et al. (2005). Promoter methylation of dal-1/4.1 b predicts poor prognosis in non–small cell lung cancer. *Clin. Cancer Res.* 11, 2954–2961. doi: 10.1158/1078-0432.CCR-04-2206

Koo, B.-H., Hurskainen, T., Mielke, K., Aung, P. P., Casey, G., Autio-Harmainen, H., et al. (2007). Adamtsl3/punctin-2, a gene frequently mutated in colorectal tumors, is widely expressed in normal and malignant epithelial cells, vascular endothelial cells and other cell types, and its mrna is reduced in colon cancer. *Int. J. Cancer* 121, 1710–1716. doi: 10.1002/ijc.22882

Kumar, P., Henikoff, S., and Ng, P. C. (2009). Predicting the effects of coding non-synonymous variants on protein function using the sift algorithm. *Nat. Protoc.* 4:1073. doi: 10.1038/nprot.2009.86

Lawrence, M. S., Stojanov, P., Mermel, C. H., Robinson, J. T., Garraway, L. A., Golub, T. R., et al. (2014). Discovery and saturation analysis of cancer genes across 21 tumour types. *Nature* 505:495. doi: 10.1038/nature12912

Lee, S., Stewart, S., Nagtegaal, I., Luo, J., Wu, Y., Colditz, G., et al. (2012). Differentially expressed genes regulating the progression of ductal carcinoma in situ to invasive breast cancer. *Cancer Res.* 72, 4574–4586. doi: 10.1158/0008-5472.CAN-12-0636

Li, S., Mei, Z., Hu, H.-B., and Zhang, X. (2018). The lncrna malat1 contributes to non-small cell lung cancer development via modulating mir-124/stat3 axis. *J. Cell. Physiol.* 233, 6679–6688. doi: 10.1002/jcp.26325

Loh, X. Y., Ding, L. W., Koeffler, H. P. (2017). "Tumor suppressive role of ZFP36L1 by suppressing HIF1α and Cyclin D1 in bladder and breast cancer," in *AACR Annual Meeting 2017* (Washington, DC: AACR). doi: 10.1158/1538-7445.AM2017-4494

Luo, P., Tian, L.-P., Ruan, J., and Wu, F.-X. (2017). "Disease gene prediction by integrating ppi networks, clinical rna-seq data and omim data," in *IEEE/ACM Transactions on Computational Biology and Bioinformatics* (Shenzhen).

Meriggi, F., Vermi, W., Bertocchi, P., and Zaniboni, A. (2014). The emerging role of nras mutations in colorectal cancer patients selected for anti-egfr therapies. *Rev. Recent Clin. Trials* 9, 8–12. doi: 10.2174/156802661466614042312 1525

Mularoni, L., Sabarinathan, R., Deu-Pons, J., Gonzalez-Perez, A., and López-Bigas, N. (2016). Oncodrivefml: a general framework to identify coding and non-coding regions with cancer driver mutations. *Genome Biol.* 17:128. doi: 10.1186/s13059-016-0994-0

Pachter, L. (2011). Models for transcript quantification from rna-seq. *arXiv[Preprint].arXiv:1104.3889*

Pécuchet, N., Laurent-Puig, P., Mansuet-Lupo, A., Legras, A., Alifano, M., Pallier, K., et al. (2017). Different prognostic impact of stk11 mutations in non-squamous non-small-cell lung cancer. *Oncotarget* 8:23831. doi: 10.18632/oncotarget.6379

Prévostel, C., and Blache, P. (2017). The dose-dependent effect of sox9 and its incidence in colorectal cancer. *Eur. J. Cancer* 86, 150–157. doi: 10.1016/j.ejca.2017.08.037

Rao, R. C., and Dou, Y. (2015). Hijacked in cancer: the kmt2 (mll) family of methyltransferases. *Nat. Rev. Cancer* 15:334. doi: 10.1038/nrc3929

Redig, A. J., Capelletti, M., Dahlberg, S. E., Sholl, L. M., Mach, S. L., Fontes, C., et al. (2016). Clinical and molecular characteristics of nf1 mutant lung cancer. *Clin. Cancer Res.* 22, 3148–3156. doi: 10.1158/1078-0432.CCR-15-2377

Reimand, J., and Bader, G. D. (2013). Systematic analysis of somatic mutations in phosphorylation signaling predicts novel cancer drivers. *Mol. Syst. Biol.*9:637. doi: 10.1038/msb.2012.68

Rubio-Perez, C., Tamborero, D., Schroeder, M. P., Antolín, A. A., Deu-Pons, J., Perez-Llamas, C., et al. (2015). In silico prescription of anticancer drugs to cohorts of 28 tumor types reveals targeting opportunities. *Cancer Cell* 27, 382–396. doi: 10.1016/j.ccell.2015.02.007

Sanz-Pamplona, R., Lopez-Doriga, A., Paré-Brunet, L., Lázaro, K., Bellido, F., Alonso, M. H., et al. (2015). Exome sequencing reveals amer1 as a frequently mutated gene in colorectal cancer. *Clin. Cancer Res.* 21, 4709–4718. doi: 10.1158/1078-0432.CCR-15-0159

Tamborero, D., Gonzalez-Perez, A., and Lopez-Bigas, N. (2013). Oncodriveclust: exploiting the positional clustering of somatic mutations to identify cancer genes. *Bioinformatics* 29, 2238–2244. doi: 10.1093/bioinformatics/btt395

Tokheim, C. J., Papadopoulos, N., Kinzler, K. W., Vogelstein, B., and Karchin, R. (2016). Evaluating the evaluation of cancer driver genes. *Proc. Natl. Acad. Sci. U.S.A.* 113, 14330–14335. doi: 10.1073/pnas.1616440113

Uusitalo, E., Kallionpää, R. A., Kurki, S., Rantanen, M., Pitkäniemi, J., Kronqvist, P., et al. (2017). Breast cancer in neurofibromatosis type 1: overrepresentation of unfavourable prognostic factors. *Br. J. Cancer* 116:211. doi: 10.1038/bjc.2016.403

Velmurugan, K., Varghese, R., Fonville, N., and Garner, H. (2017). High-depth, high-accuracy microsatellite genotyping enables precision lung cancer risk classification. *Oncogene* 36:6383. doi: 10.1038/onc.2017.256

Vogelstein, B., Papadopoulos, N., Velculescu, V. E., Zhou, S., Diaz, L. A., and Kinzler, K. W. (2013). Cancer genome landscapes. *Science* 339:1546–1558. doi: 10.1126/science.1235122

Wang, H., Shen, Q., Zhang, X., Yang, C., Cui, S., Sun, Y., et al. (2017). The long non-coding rna xist controls non-small cell lung cancer proliferation and invasion by modulating mir-186-5p. *Cell. Physiol. Biochem.* 41, 2221–2229. doi: 10.1159/000475637

Wang, X.-W., and Zhang, Y.-J. (2014). Targeting mtor network in colorectal cancer therapy. *World J. Gastroenterol.* 20:4178. doi: 10.3748/wjg.v20.i15.4178

Wong, W. C., Kim, D., Carter, H., Diekhans, M., Ryan, M. C., and Karchin, R. (2011). Chasm and snvbox: toolkit for detecting biologically important single nucleotide mutations in cancer. *Bioinformatics* 27, 2147–2148. doi: 10.1093/bioinformatics/btr357

Yates, B., Braschi, B., Gray, K. A., Seal, R. L., Tweedie, S., and Bruford, E. A. (2016). Genenames. org: the hgnc and vgnc resources in 2017. *Nucleic Acids Res.* 45, D619–D625. doi: 10.1093/nar/gkw1033

Yu, J., Wu, W., Liang, Q., Zhang, N., He, J., Li, X., et al. (2016). Disruption of ncoa2 by recurrent fusion with lactb2 in colorectal cancer. *Oncogene* 35:187. doi: 10.1038/onc.2015.72

Yu, Y., Liu, D., Liu, Z., Li, S., Ge, Y., Sun, W., et al. (2018). The inhibitory effects of col1a2 on colorectal cancer cell proliferation, migration, and invasion. *J. Cancer* 9:2953. doi: 10.7150/jca.25542

Zheng, X., Zhou, C., Cheng, H., Hu, T., Liu, H., Liu, X., et al. (2017). ELMO1 promotes metastasis in colorectal cancer cells via activation of MAPK/ERK signaling pathway. *Cancer Res.* 77(13 Suppl.):4849. doi: 10.1158/1538-7445.AM2017-4849

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

NOMENCLATURE

Resource Identification Initiative

Genomic Data Commons Data Portal (GDC Data Portal), RRID:SCR_014514

COSMIC-Catalog Of Somatic Mutations In Cancer, RRID:SCR_002260

HGNC, RRID:SCR_002827

tensorflow, RRID:SCR_016345

frontiers in Plant Science

METHODS
published: 15 February 2019
doi: 10.3389/fpls.2019.00155

Deep Learning-Based Segmentation and Quantification of Cucumber Powdery Mildew Using Convolutional Neural Network

Ke Lin[1], Liang Gong[1]*, Yixiang Huang[1], Chengliang Liu[1]* and Junsong Pan[2]

[1] School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai, China, [2] School of Agriculture and Biology, Shanghai Jiao Tong University, Shanghai, China

OPEN ACCESS

Edited by:
Minjun Chen,
National Center for Toxicological Research (FDA), United States

Reviewed by:
Jayme Garcia Arnal Barbedo,
Brazilian Agricultural Research Corporation (EMBRAPA), Brazil
Chaoyang Zhang,
University of Southern Mississippi, United States
Mingzhen Zhang,
National Institutes of Health (NIH), United States

***Correspondence:**
Liang Gong
gongliang_mi@sjtu.edu.cn
Chengliang Liu
profchlliu@163.com

Specialty section:
This article was submitted to Bioinformatics and Computational Biology, a section of the journal Frontiers in Plant Science

Received: 01 September 2018
Accepted: 29 January 2019
Published: 15 February 2019

Citation:
Lin K, Gong L, Huang Y, Liu C and Pan J (2019) Deep Learning-Based Segmentation and Quantification of Cucumber Powdery Mildew Using Convolutional Neural Network. Front. Plant Sci. 10:155. doi: 10.3389/fpls.2019.00155

Powdery mildew is a common disease in plants, and it is also one of the main diseases in the middle and final stages of cucumber (*Cucumis sativus*). Powdery mildew on plant leaves affects the photosynthesis, which may reduce the plant yield. Therefore, it is of great significance to automatically identify powdery mildew. Currently, most image-based models commonly regard the powdery mildew identification problem as a dichotomy case, yielding a true or false classification assertion. However, quantitative assessment of disease resistance traits plays an important role in the screening of breeders for plant varieties. Therefore, there is an urgent need to exploit the extent to which leaves are infected which can be obtained by the area of diseases regions. In order to tackle these challenges, we propose a semantic segmentation model based on convolutional neural networks (CNN) to segment the powdery mildew on cucumber leaf images at pixel level, achieving an average pixel accuracy of 96.08%, intersection over union of 72.11% and Dice accuracy of 83.45% on twenty test samples. This outperforms the existing segmentation methods, K-means, Random forest, and GBDT methods. In conclusion, the proposed model is capable of segmenting the powdery mildew on cucumber leaves at pixel level, which makes a valuable tool for cucumber breeders to assess the severity of powdery mildew.

Keywords: powdery mildew, cucumber leaf, convolutional neural network, image segmentation, deep-learning

INTRODUCTION

Powdery mildew is a common fungal disease that mainly infects plant leaves. The hazards of powdery mildew are considerable and may affect photosynthesis (Watanabe et al., 2014). Indeed, when the disease is severe, the infected leaves will shed (Marçais and Desprez-Loustau, 2014), causing significant losses (Xia et al., 2016).

Therefore, it is particularly important to automatically recognize powdery mildew on plant leaves. A number of high-quality image-based methods have been developed to recognize diseases on plants (Mutka and Bart, 2015), including chlorophyll fluorescence imaging, hyperspectral imaging, thermal imaging and visible light imaging. Chlorophyll fluorescence emission, an invisible phenomenon, changes when plants are experiencing biotic and abiotic stresses (Baker, 2008). Thus, chlorophyll fluorescence imaging can be used to measure this trait. Hyperspectral imaging is a technique that can be used to obtain the spectrum for each pixel in the image of a scene, which has been widely used in plant breeding (Dale et al., 2013). In addition, some fungi can affect the transpiration of the leaves and affects the temperature of the surface of the leaves (Lindenthal et al., 2005). Thus, thermal imaging can be employed to measure the temperature of leaves to identify the different types of disease. Methods based on the chlorophyll fluorescence, hyperspectral, and thermal images require expensive equipment and sophisticated analysis methods. In contrast, visible-spectrum RGB images can be obtained using a large number of very accessible devices. As a result, it is possible to gather the data required by more sophisticated algorithms. Therefore, in recent years, many methods for detecting plant diseases using visible-spectrum images have been developed.

Based on the Hough transform of the image and the random forest algorithm, Wspanialy and Moussa (2016) built a detection machine vision system to detect early powdery mildew. In the field testing on a greenhouse of tomato plants, this method achieved 85% recognition accuracy. Zhang et al. (2017) had combined the shape and color features from the disease regions and used sparse representation classification to recognize diseased leaf images. The method they proposed was feasible in recognizing seven major diseases of cucumber, and it achieved 85.7% recognition accuracy in their test datasets. With the development of deep leaning in computer vision tasks, especially convolutional neural networks (CNN), researchers can achieve higher recognition accuracy in object detection and semantic segmentation tasks. Therefore, deep learning might be used in automatic plant disease identification (Barbedo, 2016). At present, there have been many studies using CNN for plant disease recognition. A plant disease classification model was developed by Sladojevic et al. (2016), which could distinguish 13 different types of plants disease including powdery mildew from the images of healthy plant leaves. Another study using CNN to classify diseases of plants was (Amara et al., 2017). They used the *LeNet* architecture to classify banana leaf diseases. In order to overcome the problem of the slow recognition speed of neural networks, Fuentes et al. (2017) proposed a real-time tomato plant disease and pests recognition model, which could recognize nine diseases including powdery mildew. There are also a number of studies using CNN to classify plant diseases, including (Mohanty et al., 2016; Wang et al., 2017; Ferentinos, 2018).

Notably, current image-based models commonly regard the powdery mildew identification problem as a dichotomy case, yielding a true or false classification assertion. However, quantitative evaluation of the disease resistance traits plays an important role in plant variety screening for breeders. Thus, there is an urgent need to exploit the extent to which the leaves are infected.

In this paper, we proposed a new deep learning scheme which represents powdery mildew infection by masked regions generated from the segmentation model. In this way, the exact severity of the disease can be obtained. Compared to the hyperspectral image-based method, the proposed method is easier to implement and does not require expensive special imaging equipment. Further, compared to methods based on visible image classification, our method is able to obtain the location of the disease regions. With this advantage, the proposed method can provide the area and shape of the disease regions. The former can be used to indicate the severity of the disease, and the latter can help with the morphological analysis of the disease regions. Our method is available under the open-source MIT License at *https://github.com/ChrisLinSJTU/segmentation-of-powdery-mildew*.

K-means is a typical unsupervised method that can be used for clustering. Zhang et al. (2017) employed K-means method to segment the disease regions in plant leaves. While, Random forest and Gradient boosting decision tree (Ke et al., 2017) are supervised learning methods that can be used to deal with classification and regression problems. Therefore, these three methods can be applied to classify the pixels in an image to segment the disease region. Consequently, we compared the proposed method to these three segmentation methods. However, compared with the deep learning-based methods, these three methods have lower model complexity, which means that the representation ability of these three methods is not as powerful as deep learning-based methods. Experimental results also showed that our method is superior to these three methods.

The rest of this paper is organized as Materials and Methods followed by Results and Discussion. In the Materials and Methods section, we collected image samples and proposed a convolutional neural network based on U-net. Fifty cucumber leaves infected with powdery mildew were collected, and the annotations of all cucumber leaf images were manually created. Thirty pairs (images and annotations) of them were used for training and twenty pairs were used for testing. Image augmentation techniques are used for better training the sematic segmentation model. To obtain a more robust model, we used a custom loss function and added a batch normalization (Ioffe and Szegedy, 2015) layer behind each convolutional layer. In the Result section, we used six metrics, including pixel accuracy, intersection over union (Long et al., 2015), Dice accuracy (Milletari et al., 2016), Recall, Precision and F_β score to show the results of the proposed model on twenty test samples. In addition, we compared these six metrics with the existing K-means, Random forest, and GBDT image segmentation methods. In the Discussion section, we discussed the importance of the proposed model and some findings in the experimental results.

MATERIALS AND METHODS

The image acquisition process is demonstrated in section "Sample Collection," and in section "Image Preprocessing, Network Structure of the Image Segmentation Model, Network Training, and Model Testing" we describe the pipeline of our method.

Sample Collection

In this paper, 50 cucumber leaves infected with powdery mildew were collected from Shanghai, China. The images of these samples were captured in a Cucumber Fruit Leaf Phenotype Automated Analysis Platform. It is an image-based cucumber phenotype platform whose shape is an 80 cm × 80 cm × 140 cm

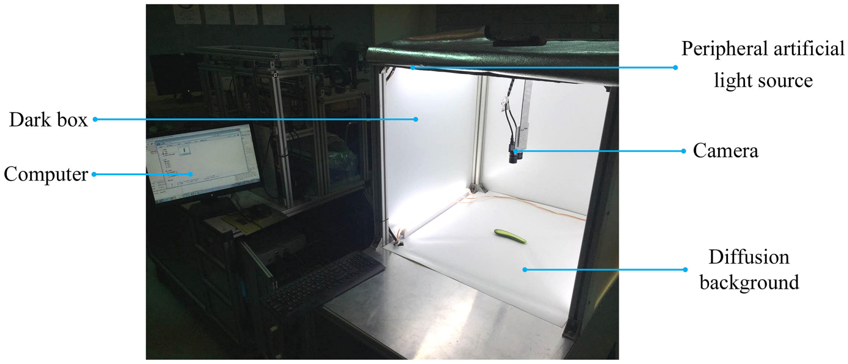


FIGURE 1 | *In vitro* Cucumber Fruit/Leaf Phenotyping platform.

rectangle. A USB camera with a resolution of 2592 × 1944 × 3 is on top of it for photographing plants. There is a diffusion background at the bottom for providing uniform illumination and a peripheral artificial light source at the top for minimizing the shadow. In addition, there is a computer next to sample holding area that is used to perform phenotypic analysis. The platform is shown in **Figure 1**. **Figure 2A** shows two samples of cucumber leaves infected with powdery mildew.

To train the CNN for identifying disease areas on the leaves, it is necessary to annotate the ground truth. Therefore, the annotations of all the cucumber leaf images were manually created. **Figure 2B** shows the disease areas of the cucumber leaves. **Figure 2C** shows the annotation images of each sample, in which the pixels of disease regions were annotated as white and the rest were annotated as black.

In these 50 images and their annotations, we randomly selected 30 pairs (images and its annotations) as a training set to train our convolutional neural network and 20 pairs as a test set to evaluate the performance of the algorithm.

Image Preprocessing

The background of the samples we collected was white, while the main feature in the powdery mildew regions is also white. Thus, it might be difficult to achieve good performance by directly using the samples with white background for training. Consequently, it is necessary to adjust the background color to black. The process of separating a leaf from image was performed with following steps: (1) an image was transformed into the HSV color space, (2) the S channel was extracted and the OTSU method (Otsu, 1979) was applied to it to obtain the mask, and (3) the RGB channels of the original picture were multiplied by the mask to obtain a picture with a black background. In addition, the images were downscaled to 512 × 512 × 3 by down-sampling.

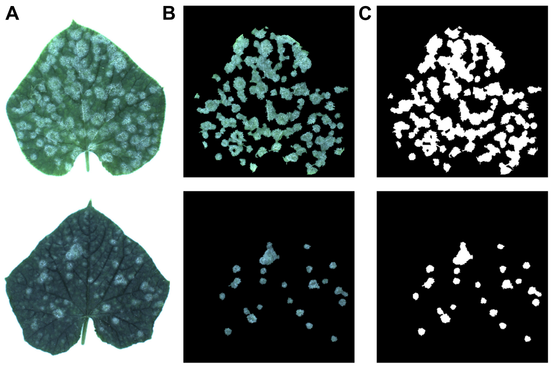


FIGURE 2 | (A) Two samples of cucumber leaves, **(B)** their disease areas, **(C)** annotation of infected areas.

Network Structure of the Image Segmentation Model

The convolutional neural network constructed in this paper is mainly based on the U-Net. U-net is one of the convolution neural networks that had shown excellent performance in biomedical image segmentation (Ronneberger et al., 2015). It is characterized by the Up-sampling layer and the concatenation of the Up-sampling layer and the previous activation layer. The process of Up-sampling makes the output of the neural network the same size as the input image, achieving pixel-level segmentation. In addition, the process of concatenation enables precise positioning of the target. These two processes are very appropriate for pixel-level segmentation of powdery mildew. Moreover, based on massive data augmentation, the network can be trained end-to-end (input is an image, and output is also an image) from very few images. This is very suitable for the agricultural field because, under normal circumstances, there are no large data sets for researcher to train neural networks, especially in the field of phenotypes. The structure of the U-net we constructed in the paper is shown in **Figure 3**.

In **Figure 3**, each color block represents a module of the neural network. The number below each color block, such as 512 × 512, represents the size of the output image of the layer. The number above each color block represents the "depth" of the current layer. In the U-net we used, the input is a color image, and the output is a grayscale image. For an output, when the pixel value is greater than 0.5, it is marked as a pixel in a disease area. Compared with the original U-net, we

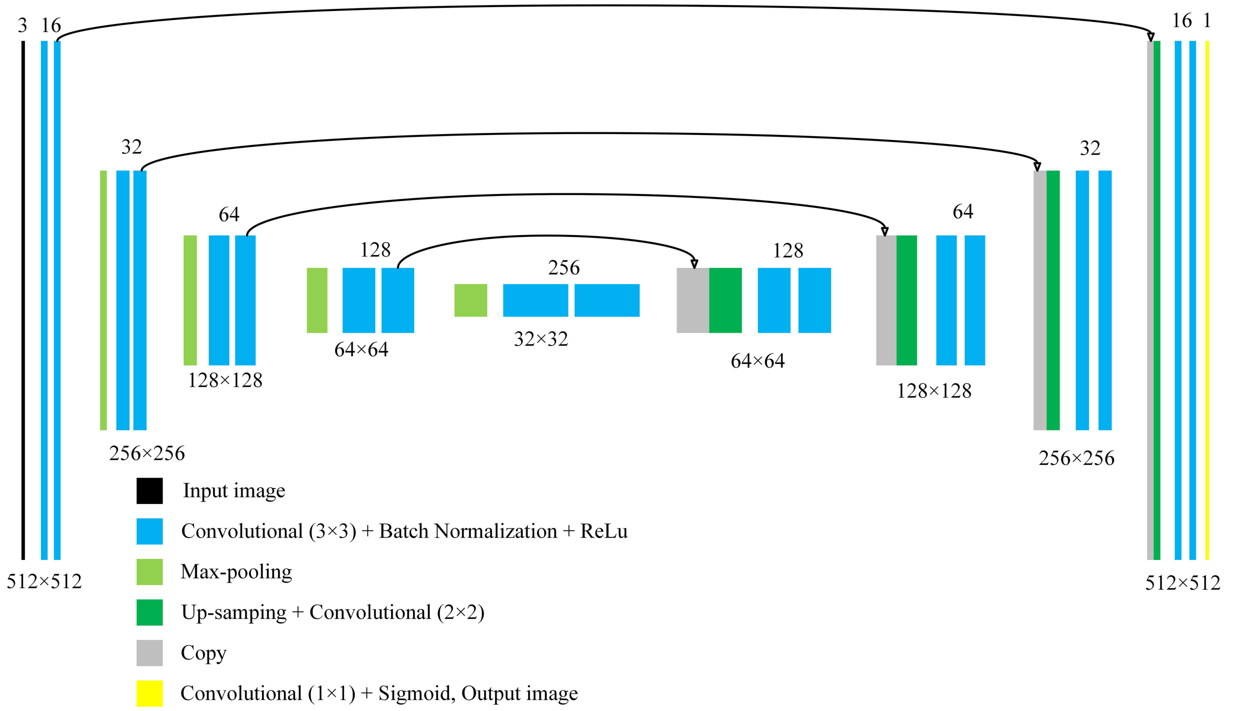

3
16
512×512
32
256×256
64
128×128
128
64×64
256
32×32
128
64×64
64
128×128
32
256×256
16
1
512×512
Input image
Convolutional (3×3) + Batch Normalization + ReLu
Max-pooling
Up-samping + Convolutional (2×2)
Copy
Convolutional (1×1) + Sigmoid, Output image

FIGURE 3 | The structure of the proposed model.

had added a batch normalization layer behind each convolution layers with a 3 × 3 convolution kernel. The addition of batch normalization allows us to use higher learning rates to accelerate the training process, and it also has the effect of regularization (Ioffe and Szegedy, 2015). In addition, after adding the batch normalization layer, the neural network becomes insensitive to weight initialization.

The segmentation of disease regions is essentially a binary classification problem which is performed on each pixel. However, the number of pixels of disease regions are smaller than non-disease regions. Thus, this creates a situation that the positive and negative samples are not balanced, which could make the neural network tend to have a low accuracy on the category with fewer samples (Huang et al., 2016). This could lead to a lower recognition accuracy in disease regions. To solve this problem, based on the binary cross entropy loss function (Goodfellow et al., 2016), we had magnified the loss value of the positive pixels by 10 times, in which the value of 10 was determined empirically. The loss function we used is shown in Eq. 1.

$$L = \sum_{i=1}^{m} -(10 \times y_i \times \log\left(y_i^{'}\right) + (1 - y_i) \times \log(1 - y_i^{'})) \qquad (1)$$

m denotes the number of pixels in an image. y_i denotes the real value of the i-th pixel, whose value is 0 or 1. $y_i^{'}$ denotes the predicted value of the i-th pixel by the method, whose range is 0 to 1.

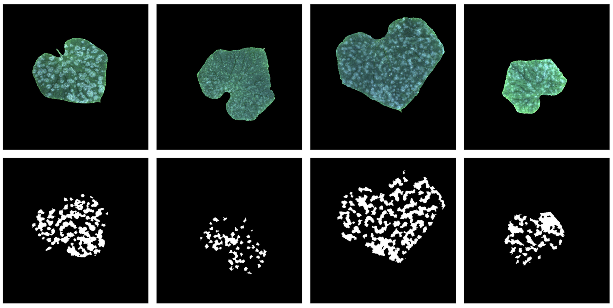


FIGURE 4 | Image augmentation of four samples (images and their annotation).

Network Training

Since the training sample has only 30 images, we had to expand these 30 images to train the neural network more effectively. Expansion methods include rotation, horizontal and vertical shift, zooming in and zooming out, horizontal flipping and vertical flipping. The range of rotation is 0 to 180 degrees, and the range of horizontal and vertical shift is 0.1 times width and height of the image, respectively; the zoom range is 0.6 to 1.4. The values of the four transformations to an image are all randomly selected from their range. Moreover, when an image was transformed, its annotation image was also transformed in the same way. In addition, since the parameters of the transformations are randomly selected, it is necessary to generate a random number.

TABLE 1 | Accuracy of our model and K-means method in 20 test samples*.

No.	Our model IU acc.	Our model Dice acc.	Our model Pixel acc.	K-means IU acc.	K-means Dice acc.	K-means Pixel acc.
1	69.93%	82.31%	97.76%	36.07%	53.02%	93.55%
2	81.92%	90.06%	95.65%	46.96%	63.91%	89.17%
3	53.98%	70.12%	99.24%	14.55%	25.41%	96.64%
4	83.41%	90.95%	94.73%	44.89%	61.96%	84.93%
5	82.35%	90.32%	96.88%	66.46%	79.85%	94.75%
6	73.04%	84.42%	96.17%	57.79%	73.25%	94.73%
7	82.68%	90.52%	95.78%	62.88%	77.21%	92.01%
8	83.11%	90.77%	95.60%	49.55%	66.26%	88.34%
9	63.33%	77.55%	96.79%	40.80%	57.95%	95.61%
10	71.71%	83.53%	96.67%	51.00%	67.55%	94.79%
11	73.00%	84.40%	96.43%	58.14%	73.53%	95.24%
12	79.20%	88.39%	96.98%	59.01%	74.22%	94.50%
13	64.31%	78.28%	97.76%	39.77%	56.91%	95.72%
14	85.65%	92.27%	94.42%	45.33%	62.38%	81.33%
15	65.78%	79.36%	93.18%	45.82%	62.84%	90.47%
16	67.21%	80.39%	95.14%	46.27%	63.27%	92.79%
17	54.09%	70.20%	95.34%	32.46%	49.01%	91.85%
18	72.71%	84.20%	93.90%	51.34%	67.85%	91.20%
19	64.99%	78.78%	96.33%	49.41%	66.14%	95.27%
20	69.76%	82.19%	96.80%	42.51%	59.65%	93.76%

**The average accuracy values of IU, Dice, Pixel of the proposed model in test samples are 72.11, 83.45, and 96.08%, respectively; the average accuracy values of IU, Dice, Pixel of the K-means method in test samples are 47.05, 63.11, and 92.33%, respectively.*

To ensure the generated data is the same in each epoch during the training process, we fixed the value of the random seed as 1. Based on 30 training samples and transformation methods, 10,000 training data pairs were generated. Four generated images and the correspond annotation images are show in **Figure 4**.

In the optimization process, the Adam method was applied with the learning rate of 0.0001, and other parameters are consistent with those in the original manuscript (Kingma and Ba, 2014). As for the initialization of the weights, we used the Glorot initialization method (Glorot and Bengio, 2010). We trained our model with the generated 10,000 pairs, where the batch size for each iteration was 2 with 32 epochs.

The hardware used for training the model is a GPU server equipped with an Intel Xeon E5-2620 CPU and an NVIDIA TESLA P100 GPU. We implemented our model with a high-level neural network API called Keras (Chollet, 2015) with the Tensorflow (Abadi et al., 2016) backend running on the Ubuntu 16.04 operating system.

Model Testing

Pixel-level segmentation of images is also known as semantic segmentation, in which the common metrics include pixel accuracy, intersection over union (Long et al., 2015) and dice accuracy (Milletari et al., 2016). The equations of these three metrics are shown in Eqs 2, 3, and 4, where p_{tf} denotes the number of pixels which are marked as disease regions by both the output of the algorithm and the ground truth in an image; p_t and p_f denote the number of pixels which are marked as disease regions by the ground truth and the output of the algorithm, respectively. In this paper, we use these three metrics to assess the performance of the method.

$$Acc_{Pixel} = \frac{1}{m}\sum_{i=1}^{m} f_i, \quad f_i = \begin{cases} 1 & y_i = y_i^{'} \\ 0 & y_i \neq y_i^{'} \end{cases} \tag{2}$$

$$Acc_{IU} = \frac{p_{tf}}{p_t + p_f - p_{tf}} \tag{3}$$

$$Acc_{Dice} = \frac{2 \times p_{tf}}{p_t + p_f} \tag{4}$$

To verify the performance of the proposed model, we used 20 samples to test it. The three metrics mentioned above, IU accuracy, Dice accuracy and Pixel accuracy, were used to evaluate the performance of the model. Since the final output of our model is a 512 × 512 grayscale image and the values of all pixels vary

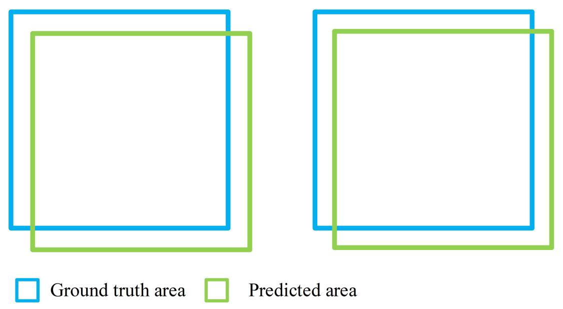


FIGURE 5 | Situation when Dice acc and IU acc are 0.8 (left) and 0.7 (right).

from 0 to 1, a threshold, whose value is 0.5, was set to binarize the output to obtain the segmented region. Recall, Precision and F_β (Powers, 2011) were also used to evaluate the performance of the model. Generally, for disease recognition, all disease areas are supposed to be detected by the algorithm. As a consequence, Recall usually has priority over Precision. So, we set the β in F_β as 2, which means the Recall is twice as important as the Precision.

Zhang et al. (2017) applied a sparse representation classification method to recognize multiple diseases on cucumber leaves, in which the K-means method was employed to segment the disease regions. Therefore, we also compared our model with the K-means disease segmentation method in detail.

RESULTS

Results of 20 Test Samples

Our model achieved satisfactory segmentation accuracy on 20 test samples. The result of IU accuracy, Dice accuracy and Pixel accuracy of the proposed model and the K-means method are shown in **Table 1**. Our models performed better than the K-means method on these three metrics. The average IU, Dice and Pixel accuracy of the former are 72.11, 83.45, and 96.08%, respectively, while the latter are 47.05, 63.11, and 92.33%, respectively. Generally, in the same segmentation performance, the value of Dice accuracy is usually greater than IU accuracy. For Dice accuracy, 0.8 can be a good value, while 0.7 is good for IU accuracy. **Figure 5** shows the situation when Dice accuracy and IU accuracy are 0.8 and 0.7, respectively, in which they almost have the same segmentation performance.

The results of Precision, Recall and F_2-score of our model and K-means method are shown in **Table 2**. The average Precision, Recall and F_2-score of the former are 73.30, 97.34, and 91.20%, respectively, while the latter are 71.35, 60.55, and 60.83%, respectively. The precision of the proposed model is not very good, but the recall is quite high, which means the model has a certain degree of over-segmentation. A further explanation is that the most disease regions had been recognized; however, some non-disease areas had been misidentified as disease areas. This

TABLE 2 | Precision, Recall and F-score of our model and K-means method*.

No.	Our model Precision	Our model Recall	Our model F2 Score	K-means Precision	K-means Recall	K-means F2 Score
1	70.69%	98.48%	91.30%	43.08%	68.92%	61.54%
2	82.10%	99.74%	95.63%	93.61%	48.52%	53.69%
3	56.90%	91.32%	81.46%	16.20%	58.86%	38.56%
4	83.57%	99.77%	96.04%	94.00%	46.21%	51.44%
5	82.80%	99.34%	95.52%	91.17%	71.04%	74.32%
6	73.86%	98.50%	92.34%	78.70%	68.51%	70.33%
7	83.10%	99.39%	95.64%	91.50%	66.78%	70.60%
8	83.55%	99.37%	95.74%	89.44%	52.63%	57.35%
9	64.47%	97.30%	88.31%	63.78%	53.10%	54.94%
10	72.42%	98.66%	91.99%	72.42%	63.30%	64.93%
11	73.32%	99.42%	92.81%	79.89%	68.11%	70.18%
12	79.50%	99.53%	94.76%	81.04%	68.47%	70.66%
13	66.39%	95.35%	87.70%	49.58%	66.79%	62.45%
14	85.88%	99.68%	96.58%	95.43%	46.34%	51.65%
15	71.35%	89.38%	85.08%	73.48%	54.89%	57.82%
16	68.33%	97.63%	89.91%	65.84%	60.89%	61.82%
17	59.08%	86.48%	79.14%	40.65%	61.70%	55.90%
18	73.13%	99.22%	92.61%	84.52%	56.67%	60.67%
19	65.46%	98.89%	89.72%	65.23%	67.06%	66.69%
20	70.04%	99.44%	91.74%	57.37%	62.13%	61.12%

The average Precision, Recall and F2 score of our model in test samples are 73.30, 97.34, and 91.20%, respectively; the average Precision, Recall and F2 score of K-means method in test samples are 71.35, 60.55, and 60.83%, respectively.

TABLE 3 | The performance of our method and the three other methods*.

Method	Precision	Recall	F2 score	IU acc.	Dice acc.	Pixel acc.
The proposed method	73.30%	97.34%	91.20%	72.11%	83.45%	96.08%
GBDT	73.90%	70.81%	70.86%	56.96%	71.44%	94.33%
Random Forest	70.99%	69.33%	69.20%	54.84%	69.46%	93.95%
K-means	71.35%	60.55%	60.83%	47.05%	63.11%	92.33%

The average accuracy of Precision, Recall, F2 score, IU accuracy, Dice accuracy, and Pixel accuracy of the proposed model and three other methods.

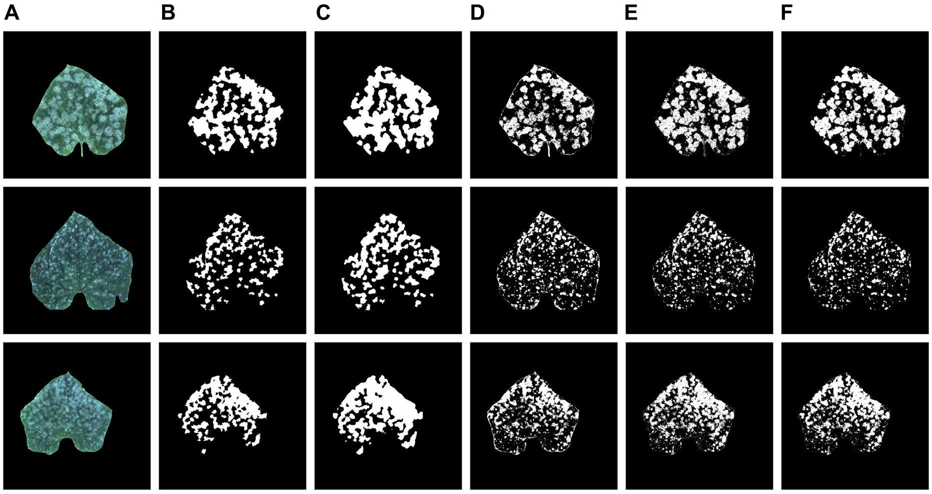


FIGURE 6 | (A) Original images, **(B)** annotation images, **(C–F)** recognition results of the proposed model, K-means, Random forest, and GBDT methods.

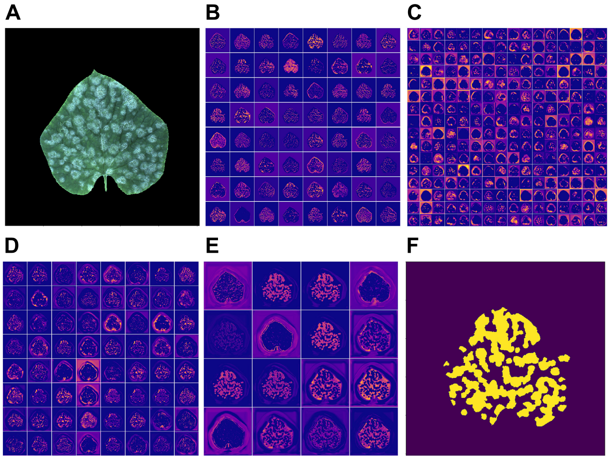


FIGURE 7 | (A) Input image; **(B–E)** feature map of the proposed model given this input image; **(F)** output image.

situation is acceptable, because, for disease detection, the disease regions are not supposed to be missed by the algorithm.

In addition, we also compared the proposed method to the Random forest method and GBDT (Ke et al., 2017) method. Although these two methods are supervised learning method, usually used for classification and regression, they also can be used to image segmentation regarding pixels as classification targets. As above, 30 images were used for training and 20 images were used for testing. Each image contains 262,144 pixels (512 × 512), so the training set contains a total of 7,861,320 samples. Testing set contains 5242,880 samples. Lightgbm (Ke et al., 2017) and scikit-learn (Pedregosa et al., 2011), two Python packages, were used to implement these two methods separately. The results show that the proposed methods have the best performance in terms of IU accuracy, Dice accuracy, Pixel accuracy, and Recall in twenty test images. However, for the metric of Precision, the average accuracy of our method is slightly lower than GBDT. These can be seen in **Table 3**.

Output of 3 Samples by Proposed Model

Figure 6 shows the recognition results of the proposed model, K-means, Random forest, and GBDT methods on three test samples, which include the original images, the annotation image, the segmentation results of the proposed model and the segmentation results of the other three methods. As can be seen in **Figure 6**, when compared to the annotation images, the prediction results of the proposed model have greater predicted areas, which is consistent with the relatively high Recall.

As for the prediction result of the K-means, Random forest, and GBDT methods, the areas of the segmentation are relatively small. Thus, it leads to a unilateral bias of under segmentation of the infected disease regions, which is evident in **Figures 6D–F**.

Visualization of the Feature Map of CNN Model

Feature map opens the gray box of a deep-learning based model, illustrating the intermediate result of the learning process. **Figure 7** shows the feature map of the middle layers and an output image (f) produced by the network when given the input image (a). **Figures 7B–D** show the output of the activation layer after the sixth, tenth, and fourteenth convolutional layers, respectively. **Figure 7E** shows the output of the last activation layer.

As can be seen from the **Figure 7B**, the edge of the leaf the disease regions are highlighted by the convolutional neural network. When it comes to the output of the middle layer which is shown in **Figure 7C**, the feature map appears to be more abstract. This is because the middle layer of a neural network is difficult to interpret in general. In **Figure 7D**, the output of the convolutional neural network has no obvious sharp edges. The edges of the leaf gradually fade, which is expected because the model is supposed to pay more attention to the disease region

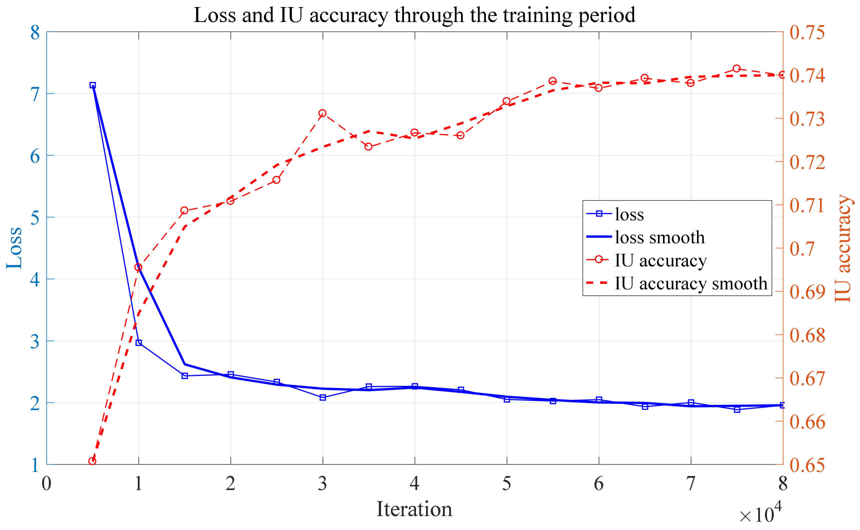


FIGURE 8 | Loss and IU accuracy through the training period.

rather the edge of a leaf. In the output of the activation layer shown in **Figure 7E**, which is close to the output layer, the disease region becomes more concentrated. **Figure 8** shows the convergence process of the loss function value and IU accuracy of the proposed segmentation model during the period of training, in which the bold line is the result of smoothing the original curve for better demonstration.

DISCUSSION

This study aimed to tackle the problem of segmenting powdery mildew on leaves accurately based on visible images. To address this problem, we proposed a convolutional neural network model based on the U-net architecture which is used for sematic segmentation tasks in the field of computer vision. Experimental results on 20 test samples demonstrated that, compared to the existing K-means, Random forest, and GBDT image segmentation methods, the proposed method greatly improved the accuracy of powdery mildew segmentation. However, the proposed method may have greater computational complexity, which means it might be hard to deploy the proposed method to portable device.

Compared to some feature-based plant disease identification methods, this method alleviates researchers from manually extracting complex features in the image and designing complicated analytical methods. In addition, compared with some existing methods based on deep learning for classifying and identifying disease on plant leaves, our method can segment powdery mildew on a cucumber leaf at the pixel level. In summary, the principal discoveries include:

1. In twenty test samples, our model achieved a satisfactory segmentation accuracy of powdery mildew under three metrics of IU accuracy, Dice accuracy and Pixel accuracy. Moreover, the Pixel accuracy of all samples is relatively high, which means that the performance of the proposed model when segmenting powdery mildew on cucumber leaves is feasible in practice. We also randomly selected three samples from twenty test samples to compare the output of the proposed model and the three other methods. The mask image output by the proposed model had a certain degree of over-segmentation when segmenting powdery mildew. However, the mask image obtained by the K-means method had a certain degree of under-segmentation. In addition, the edges of the predicted area of the proposed model were smoother than the K-means method. Generally speaking, the regions of powdery mildew usually appear in block form. Therefore, the smoother edge of the disease region is expected.
2. Unbalanced positive and negative samples in the image cause relatively high segmentation accuracy, in which there are more pixels belonging to the background. Furthermore, the background might be easier to be recognized than the foreground.

In addition, we also found an interesting phenomenon where, in some test samples (such as sample number 3), the Pixel accuracy is high, while the IU accuracy and Dice accuracy are relatively low. After analyzing the image of this sample and the output mask of the K-means algorithms, we found that the area of the disease region in the image was very small. Since the non-disease area is easier to identify, the Pixel accuracy is very high in sample number 3. On the metric of Recall, our model achieved good accuracy on these twenty samples. In general, Recall and Precision are a pair of contradictory metrics. Higher Recall typically corresponds to lower Precision, which explains the paradox of segmentation of powdery mildew in cucumber leaves. In general, higher Recall is preferred because it can lead to the production of models which miss less disease regions. Analysis and experimentation reveal that the proposed convolutional neural network based on the U-net can segment powdery mildew on cucumber leaves accurately at pixel level and can improve on the segmentation accuracy of the existing methods. The improvement of segmentation accuracy helps to estimate the severity of powdery mildew on leaves more accurately, which makes our improved software a valuable tool for cucumber breeders.

However, it is worth noting that there are some limitations in this method. Given the fact that the images are collected on our platform, to implement the proposed method, the images need to be captured under controlled conditions, not in the field. In addition, the insufficient size and variety of annotated datasets, in which symptoms caused by other disorders are not contained in our dataset, may be a factor that influences the performance of deep learning methods (Barbedo, 2018). Thus, other types of leaf damage should be minimal absent.

AUTHOR CONTRIBUTIONS

KL, LG, and YH conducted mathematical modeling and article writing. KL also completed the software development and experimental verification. CL and JP supervised the whole project and conducted the experimental verification.

FUNDING

This research was funded by the Agri-tech Program of Shanghai Science and Technology Committee under Grant No. 16391903101 and by the Foundation of Key Laboratory of Urban Agriculture in South China, Ministry of Agriculture, China under Grant No. 2017-009. This research was also funded by the 2015–2017 Shanghai Jiao Tong University "Agri+X" Funding, "establishment and application of automated analysis platform for cucumber complex phenotype" (Agri-X2015002).

ACKNOWLEDGMENTS

The help of Jingwei Zhang in sample collection and Zhihong Ma in construction of the experimental platform are gratefully acknowledged.

REFERENCES

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., et al. (2016). "Tensorflow: a system for large-scale machine learning," in *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation*. Savannah, GA 265–283.

Amara, J., Bouaziz, B., and Algergawy, A. (2017). "A deep learning-based approach for banana leaf diseases classification," in *BTW Workshops*, (Bonn: Lecture Notes in Informatics (LNI), Gesellschaft für Informatik), 79–88.

Baker, N. R. (2008). Chlorophyll fluorescence: a probe of photosynthesis in vivo. *Annu. Rev. Plant Biol.* 59, 89–113. doi: 10.1146/annurev.arplant.59.032607.092759.

Barbedo, J. G. A. (2018). Factors influencing the use of deep learning for plant disease recognition. *Biosyst. Eng.* 172, 84–91. doi: 10.1016/j.biosystemseng.2018.05.013.

Barbedo, J. G. A. (2016). A review on the main challenges in automatic plant disease identification based on visible range images. *Biosyst. Eng.* 144, 52–60. doi: 10.1016/j.biosystemseng.2016.01.017.

Chollet, F. (2015). *Keras: Deep Learning Library for Theano and Tensorflow*. Available at: https://keras.io/ [accessed on August 31, 2018].

Dale, L. M., Thewis, A., Boudry, C., Rotar, I., Dardenne, P., Baeten, V., et al. (2013). Hyperspectral imaging applications in agriculture and agro-food product quality and safety control: a review. *Appl. Spectrosc. Rev.* 48, 142–159. doi: 10.1080/05704928.2012.705800.

Ferentinos, K. P. (2018). Deep learning models for plant disease detection and diagnosis. *Comput. Electron. Agric.* 145, 311–318. doi: 10.1016/j.compag.2018.01.009.

Fuentes, A., Yoon, S., Kim, S. C., and Park, D. S. (2017). A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. *Sensors* 17:2022. doi: 10.3390/s17092022.

Glorot, X., and Bengio, Y. (2010). "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, Italy, 249–256.

Goodfellow, I., Bengio, Y., Courville, A., and Bengio, Y. (2016). *Deep Learning*. Cambridge, MA: MIT press.

Huang, C., Li, Y., Change Loy, C., and Tang, X. (2016). Learning deep representation for imbalanced classification in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, (Berlin: Springer), 5375–5384.

Ioffe, S., and Szegedy, C. (2015). Batch normalization: accelerating deep network training by reducing internal covariate shift in *Proceedings of the 32nd International Conference on International Conference on Machine Learning*, (Washington, DC: JMLR.org), 448–456.

Kingma, D. P., and Ba, J. L. (2014). Adam: a method for stochastic optimization. *arXiv* [Preprint]. arXiv:1412.6980v9

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., et al. (2017). "Lightgbm: a highly efficient gradient boosting decision tree," in *Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017)*, Long Beach, CA, 3146–3154.

Lindenthal, M., Steiner, U., Dehne, H. W., and Oerke, E. C. (2005). Effect of downy mildew development on transpiration of cucumber leaves visualized by digital infrared thermography. *Phytopathology* 95, 233–240. doi: 10.1094/PHYTO-95-0233.

Long, J., Shelhamer, E., and Darrell, T. (2015). Fully convolutional networks for semantic segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* 39:4. doi: 10.1109/TPAMI.2016.2572683.

Marçais, B., and Desprez-Loustau, M. -L. (2014). European oak powdery mildew: impact on trees, effects of environmental factors, and potential effects of climate change. *Ann. Forest Sci.* 71, 633–642.

Milletari, F., Navab, N., and Ahmadi, S. (2016). "V-net: fully convolutional neural networks for volumetric medical image segmentation," in *2016 Fourth International Conference on 3D Vision, IEEE* Hoboken, NJ, 565–571. doi: 10.1109/3DV.2016.79.

Mohanty, S. P., Hughes, D. P., and Salathé, M. (2016). Using deep learning for image-based plant disease detection. *Front. Plant Sci.* 7:1419 doi: 10.3389/fpls.2016.01419.

Mutka, A. M., and Bart, R. S. (2015). Image-based phenotyping of plant disease symptoms. *Front. Plant Sci.* 5:734. doi: 10.3389/fpls.2014.00734.

Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Trans. Syst. Man Cybern. Syst.* 9, 62–66. doi: 10.1109/TSMC.1979.4310076.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). Scikit-learn: machine learning in python. *J. Mach. Learn. Res.* 12, 2825–2830.

Powers, D. M. (2011). Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. *J. Mach. Learn. Technol.* 2, 37–63.

Ronneberger, O., Fischer, P., and Brox, T. (2015). "U-net: convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Berlin: Springer, 234–241. doi: 10.1007/978-3-319-24574-4_28

Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D., and Stefanovic, D. (2016). Deep neural networks based recognition of plant diseases by leaf image classification. *Comput. Intell. Neurosci.* 2016:1–11. doi: 10.1155/2016/3289801.

Wang, G., Sun, Y., and Wang, J. (2017). Automatic image-based plant disease severity estimation using deep learning. *Comput. Intell. Neurosci.* 2017:1–8. doi: 10.1155/2017/2917536.

Watanabe, M., Kitaoka, S., Eguchi, N., Watanabe, Y., Satomura, T., Takagi, K., et al. (2014). Photosynthetic traits and growth of Quercus mongolica var. crispula sprouts attacked by powdery mildew under free-air CO2 enrichment. *Eur. J. Forest Res.* 133, 725–733. doi: 10.1007/s10342-013-0744-8.

Wspanialy, P., and Moussa, M. (2016). Early powdery mildew detection system for application in greenhouse automation. *Comput. Electron. Agric.* 127, 487–494. doi: 10.1016/j.compag.2016.06.027.

Xia, C., Li, N., Zhang, X., Feng, Y., Christensen, M. J., and Nan, Z. (2016). An Epichloë endophyte improves photosynthetic ability and dry matter production of its host achnatherum inebrians infected by *Blumeria graminis* under various soil water conditions. *Fungal Ecol.* 22, 26–34. doi: 10.1016/j.funeco.2016.04.002.

Zhang, S., Wu, X., You, Z., and Zhang, L. (2017). Leaf image based cucumber disease recognition using sparse representation classification. *Comput. Electron. Agric.* 134, 135–141. doi: 10.1016/j.compag.2017.01.014.

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

ORIGINAL RESEARCH
published: 19 February 2019
doi: 10.3389/fgene.2019.00080

Deep Learning Based Analysis of Histopathological Images of Breast Cancer

Juanying Xie[1]*, *Ran Liu*[1], *Joseph Luttrell IV*[2] *and Chaoyang Zhang*[2]*

[1] School of Computer Science, Shaanxi Normal University, Xi'an, China, [2] School of Computing Sciences and Computer Engineering, University of Southern Mississippi, Hattiesburg, MS, United States

Breast cancer is associated with the highest morbidity rates for cancer diagnoses in the world and has become a major public health issue. Early diagnosis can increase the chance of successful treatment and survival. However, it is a very challenging and time-consuming task that relies on the experience of pathologists. The automatic diagnosis of breast cancer by analyzing histopathological images plays a significant role for patients and their prognosis. However, traditional feature extraction methods can only extract some low-level features of images, and prior knowledge is necessary to select useful features, which can be greatly affected by humans. Deep learning techniques can extract high-level abstract features from images automatically. Therefore, we introduce it to analyze histopathological images of breast cancer via supervised and unsupervised deep convolutional neural networks. First, we adapted Inception_V3 and Inception_ResNet_V2 architectures to the binary and multi-class issues of breast cancer histopathological image classification by utilizing transfer learning techniques. Then, to overcome the influence from the imbalanced histopathological images in subclasses, we balanced the subclasses with Ductal Carcinoma as the baseline by turning images up and down, right and left, and rotating them counterclockwise by 90 and 180 degrees. Our experimental results of the supervised histopathological image classification of breast cancer and the comparison to the results from other studies demonstrate that Inception_V3 and Inception_ResNet_V2 based histopathological image classification of breast cancer is superior to the existing methods. Furthermore, these findings show that Inception_ResNet_V2 network is the best deep learning architecture so far for diagnosing breast cancers by analyzing histopathological images. Therefore, we used Inception_ResNet_V2 to extract features from breast cancer histopathological images to perform unsupervised analysis of the images. We also constructed a new autoencoder network to transform the features extracted by Inception_ResNet_V2 to a low dimensional space to do clustering analysis of the images. The experimental results demonstrate that using our proposed autoencoder network results in better clustering results than those based on features extracted only by Inception_ResNet_V2 network. All of our experimental results demonstrate that Inception_ResNet_V2 network based deep transfer learning provides a new means of performing analysis of histopathological images of breast cancer.

Keywords: histopathological images, breast cancer, deep convolutional neural networks, autoencoder, transfer learning, classification, clustering

OPEN ACCESS

Edited by:
Alfredo Pulvirenti,
Università degli Studi di Catania, Italy

Reviewed by:
Omar Al-Kadi,
Yale University, United States
Hamid Reza Marateb,
Universitat Politecnica de Catalunya,
Spain

****Correspondence:***
Juanying Xie
xiejuany@snnu.edu.cn
Chaoyang Zhang
chaoyang.zhang@usm.edu

Specialty section:
This article was submitted to
Bioinformatics and Computational
Biology,
a section of the journal
Frontiers in Genetics

Received: *26 September 2018*
Accepted: *28 January 2019*
Published: *19 February 2019*

Citation:
Xie J, Liu R, Luttrell IV J and Zhang C
(2019) Deep Learning Based Analysis
of Histopathological Images of Breast
Cancer. Front. Genet. 10:80.
doi: 10.3389/fgene.2019.00080

INTRODUCTION

Cancers have become one of the major public health issues. According to statistics by the IARC (International Agency for Research on Cancer) from the WHO (World Health Organization), and GBD (Global Burden of Disease Cancer Collaboration), cancer cases increased by 28% between 2006 and 2016, and there will be 2.7 million new cancer cases emerging in 2030 (Boyle and Levin, 2008; Moraga-Serrano, 2018). Among the various types of cancer, breast cancer is one of the most common and deadly in women (1.7 million incident cases, 535,000 deaths, and 14.9 million disability-adjusted life years) (Moraga-Serrano, 2018). Therefore, the diagnosis of breast cancer has become very important. Although the diagnosis of breast cancers has been performed for more than 40 years using X-ray, MRI (Magnetic Resonance Imaging), and ultrasound etc. (Stenkvist et al., 1978), biopsy techniques are still the main methods relied on to diagnose breast cancer correctly. Common biopsy techniques include fine-needle aspiration, vacuum-assisted biopsy and surgical biopsy. The process involves collecting samples of cells or tissues, fixing them on the microscope slide, and then staining them (Veta et al., 2014). After that, the histopathological images are analyzed and the diagnosis is made by pathologists (Spanhol et al., 2016a).

However, the analysis of the histopathological images is a difficult and time-consuming task that requires the knowledge of professionals. Furthermore, the outcome of the analysis may be affected by the level of experience of the pathologists involved. Therefore, computer-aided (Aswathy and Jagannath, 2017) analysis of histopathological images plays a significant role in the diagnosis of breast cancer and its prognosis. However, the process of developing tools for performing this analysis is impeded by the following challenges. First, histopathological images of breast cancer are fine-grained, high-resolution images that depict rich geometric structures and complex textures. The variability within a class and the consistency between classes can make classification extremely difficult, especially when dealing with multiple classes. The second challenge is the limitations of feature extraction methods for histopathological images of breast cancer. Traditional feature extraction methods, such as scale-invariant feature transform (SIFT) (Lowe, 1999) and gray-level co-occurrence matrix (GLCM) (Haralick et al., 1973), all rely on supervised information. Furthermore, prior knowledge of data is needed to select useful features, which makes the feature extraction efficiency very low and the computational load very heavy. In the end, the final extracted features are only some low-level and unrepresentative features of histopathological images. Consequently, this can lead to the final model producing poor classification results.

Deep learning techniques have the power to automatically extract features, retrieve information from data automatically, and learn advanced abstract representations of data. They can solve the problems of traditional feature extraction and have been successfully applied in computer vision (He et al., 2015; Xie et al., 2018), biomedical science (Gulshan et al., 2016; Esteva et al., 2017) and many other fields.

In view of the powerful feature extraction advantages of deep learning and the challenges in histopathological image analysis of breast cancer, this paper analyzes histopathological images of breast cancer using deep learning techniques. On one hand, we use advanced deep convolutional neural networks, including Inception_V3 (Szegedy et al., 2016) and Inception_ResNet_V2 (Szegedy et al., 2017), combined with transfer learning techniques to classify the histopathological images of breast cancer (Pan and Yang, 2010). On the other hand, by combining deep learning with clustering and utilizing the dimension-reduction functionality of the autoencoder network (Hinton and Salakhutdinov, 2006), we propose a new autoencoder network structure to apply non-linear transformations to features in histopathological images of breast cancer extracted by the Inception_ResNet_V2 network. This effectively maps the extracted features to a lower dimensional space. The newly obtained features are then used as input for the classical clustering algorithm known as K-means (MacQueen, 1967) to perform clustering analysis on histopathological images of breast cancer. Also, we designed a number of comparable experiments to verify the validity of our proposed method of histopathological image analysis of breast cancer images based on deep learning techniques.

RELATED WORKS

Breast cancer diagnosis based on image analysis has been studied for more than 40 years, and there have been several notable research achievements in the area. These studies can be divided into two categories according to their methods: one is based on traditional machine learning methods, and the other is based on deep learning methods. The former category is mainly focused on small datasets of breast cancer images and is based on labor intensive and comparatively low-performing, abstract features. The latter category can deal with big data and can also extract much more abstract features from data automatically.

For example, Zhang et al. (2013) proposed a new cascade random subspace ensemble scheme with rejection options for microscopic biopsy image classification in 2012. This classification system consists of two random subspace classifier ensembles. The first ensemble consists of a set of support vector machines which correspond to the K binary classification problems transformed from the original K-class classification problem (K = 3). The second ensemble consists of a Multi-Layer Perceptron ensemble which focuses on rejected samples from the first ensemble. This system was tested on a database composed of 361 images, of which 119 were normal tissue, 102 were carcinoma *in situ*, and 140 were lobular carcinoma or invasive ductal. The authors randomly split the images into training and testing sets, with 20% of each class' images used for testing and the rest used for training. It obtained a high classification accuracy of 99.25% and a high classification reliability of 97.65% with a small rejection rate of 1.94%. In 2013, Kowal et al. (2013) used four clustering algorithms to perform nuclei segmentation for 500 images from 50 patients with breast cancer. Then, they used three different classification approaches to classify these images into benign and malignant tumors. Among 500 images, there were 25 benign and 25 malignant cases with 10 images per case. They achieved classification accuracy between 96 and

100% using a 50-fold cross-validation technique. In the same year, Filipczuk et al. (2013) presented a breast cancer diagnosis system based on the analysis of cytological images of fine needle biopsies to discriminate between benign or malignant biopsies. Four traditional machine learning methods including KNN (K-nearest neighbor with K = 5), NB (naive Bayes classifier with kernel density estimate), DT (decision tree) and SVM (support vector machine with Gaussian radial basis function kernel and scaling factor $\sigma = 0.9$) were used to build the classifiers of the biopsies with 25 features of the nuclei. These classifiers were tested on a set of 737 microscopic images of fine needle biopsies obtained from 67 patients, which contained 25 benign (275 images) and 42 malignant (462 images) cases. The best reported effectiveness is up to 98.51%. In 2014, George et al. (2014) proposed a diagnosis system for breast cancer using nuclear segmentation based on cytological images. Four classification models were used, including MLP (multilayer perceptron using the backpropagation algorithm), PNN (probabilistic neural network), LVQ (learning vector quantization), and SVM. The parameters for each model can be found in Table 5 in George et al. (2014). The classification accuracy using 10-fold cross-validation is 76$\sim$94% with only 92 images, including 45 images of benign tumors and 47 images of malignant tumors. In 2016, a performance comparison was conducted by Asri et al. (2016) between four machine learning algorithms, including SVM, DT, NB and KNN, on the Wisconsin Breast Cancer dataset, which contains 699 instances (including 458 benign and 241 malignant cases). Experimental results demonstrated that SVM achieved the highest accuracy of 97.13% with 10-fold cross-validation.

However, the above breast cancer diagnosis studies focused on Whole-Slide Imaging (Zhang et al., 2013, 2014). Since the operation of Whole-Slide Imaging is complex and expensive, many studies based on this technique use small datasets and achieve poor generalization performance. To solve these problems, Spanhol et al. (2016a) published a breast cancer dataset called BreaKHis in 2016. BreaKHis contains 7,909 histopathological images of breast cancer from 82 patients. The authors used 6 different feature descriptors and 4 different traditional machine learning methods, including 1-NN (1 Nearest Neighbor), QDA (Quadratic Discriminant Analysis), RF (Random Forest), and SVM with the Gaussian kernel function, to perform binary diagnosis of benign and malignant tumors. The classification accuracy is between 80 and 85% using 5-fold cross-validation.

Although traditional machine learning methods have made great achievements in analyzing histopathological images of breast cancer and even in dealing with relatively large datasets, their performance is heavily dependent on the choice of data representation (or features) for the task they are trained to perform. Furthermore, they are unable to extract and organize discriminative information from data (Bengio et al., 2013). Deep learning methods typically are neural network based learning machines with much more layers than the usual neural network. They have been widely used in the medical field since they can automatically yield more abstract—and ultimately more useful—representations (Bengio et al., 2013). That is, they can extract the discriminative information or features from data without requiring the manual design of features by a domain expert (Spanhol et al., 2016b).

As a consequence, Spanhol et al. (2016b) classified histopathological images of breast cancer from BreaKHis using a variation of the AlexNet (Krizhevsky et al., 2012) convolutional neural network that improved classification accuracy by 4–6%. Bayramoglu et al. (2016) proposed to classify breast cancer histopathological images independently of their magnifications using CNN (convolutional neural networks). They proposed two different architectures: the single task CNN used to predict malignancy, and the multi-task CNN used to predict both malignancy and image magnification level simultaneously. Evaluations were carried out on the BreaKHis dataset, and the experimental results were competitive with the state-of-the-art results obtained from traditional machine learning methods.

However, the above studies on the BreaKHis dataset only focus on the binary classification problem. Multi-class classification studies on histopathological images of breast cancer can provide more reliable information for diagnosis and prognosis. As a result, Araújo et al. (2017) proposed a CNN based method to classify the hematoxylin and eosin stained breast biopsy images from a dataset composed of 269 images into four classes (normal tissue, benign lesion, *in situ* carcinoma and invasive carcinoma), and into two classes (carcinoma and non-carcinoma), respectively. An SVM classifier with the radial basis kernel function was trained using the features extracted by CNN. The accuracies of the SVM for the four-class and two-class classification problems are 77.8–83.3%, respectively. To realize the development of a system for diagnosing breast cancer using multi-class classification on BreaKHis, Han et al. (2017) proposed a class structure-based deep convolutional network to provide an accurate and reliable solution for breast cancer multi-class classification by using hierarchical feature representation. Using these techniques, they were able to achieve multi-class classification of breast cancer with a maximum accuracy of 95.9%. This study is important for precise treatment of breast cancer. In addition, Nawaz et al. (2018) presented a DenseNet based model for multi-class breast cancer classification to predict the subclass of the tumors. The experimental results on BreaKHis achieved the accuracy of 95.4%. After that, Motlagh et al. (2018) used the pre-trained model of ResNet_V1_152 (He et al., 2016) to perform diagnosis of benign and malignant tumors as well as diagnosis based on multi-class classification of various subtypes of histopathological images of breast cancer in BreaKHis. They were able to achieve an accuracy of 98.7–96.4% for binary classification and multi-class classification, respectively.

Although there are 7,909 histopathological images from 82 patients in BreaKHis, the number of images is far from enough for effectively using deep learning techniques. Therefore, we proposed to combine transfer learning techniques with deep learning to perform breast cancer diagnosis using the relatively small number of histopathological images (7,909) from the BreaKHis dataset.

The Inception_V3 (Szegedy et al., 2016) and Inception_ResNet_V2 (Szegedy et al., 2017) networks were proposed by Szegedy et al. (2016, 2017), respectively. In the 2012

ImageNet Large Scale Visual Recognition Challenge (ILSVRC) competition, the Inception_V3 network achieved 78.0–93.9% accuracy in top-1 and top-5 metrics, respectively, while the Inception_ResNet_V2 achieved 80.4–95.3% accuracy in the same evaluation.

One common method for performing transfer learning (Pan and Yang, 2010) involves obtaining the basic parameters for training a deep learning model by pre-training on large data sets, such as ImageNet, and then using the data set of the new target task to retrain the last fully-connected layer of the model. This process can achieve good results even on small data sets.

Therefore, we adopt two deep convolutional neural networks, specifically Inception_V3 and Inception_Resnet_V2, to study the diagnosis of breast cancer in the BreaKHis dataset via transfer learning techniques. To solve the unbalanced distribution of samples of histopathological images of breast cancer, the BreaKHis dataset was expanded by rotation, inversion, and several other data augmentation techniques. The Inception_ResNet_V2 network was chosen to conduct binary and multi-class classification diagnosis on the expanded set of histopathological breast cancer images for its better performance on the original dataset of BreaKHis compared to that of Inception_V3. The powerful feature extraction capability of the Inception_ResNet_V2 network was used to extract features of the histopathological images of breast cancer for the linear kernel SVM and 1-NN classifiers. The image features extracted by the Inception_ResNet_V2 network are also used as the input of the K-means algorithm to do clustering analysis for the BreaKHis dataset. Furthermore, a new autoencoder deep learning model is constructed to apply a non-linear transformation to the image features extracted by Inception_ResNet_V2 network in order to get the low-dimensional features of the image, and to do clustering analysis for BreaKHis dataset using the K-means algorithm.

DATA AND METHODS

Datasets

The dataset named BreaKHis used in this article was published by Spanhol et al. (2016a) in 2016. It is composed of 7,909 histopathological images from 82 clinical breast cancer patients. The database can be accessed through the link http://web.inf.ufpr.br/vri/breast-cancer-database. To save the original organization structure and molecular composition, each image was taken by a pathologist from a patient's breast tissue section using a surgical biopsy. Then, the images were collected via haematoxylin and eosin staining. Finally, the real class label was given to each image by pathologists via their observations of the images from a microscope. All the histopathological images of breast cancer are 3 channel RGB micrographs with a size of 700 × 460. Since objective lenses of different multiples were used in collecting these histopathological images of breast cancer, the entire dataset comprised four different sub-datasets, namely 40, 100, 200, and 400X. All of these sub-datasets are classified into benign and malignant tumors. Therefore, both benign and malignant tumors have four different subsets. Benign tumors include Adenosis (A), Fibroadenoma (F), Phyllodes Tumor (PT), and Tubular Adenoma (TA). Malignant tumors include Ductal Carcinoma (DC), Lobular Carcinoma (LC), Mucinous Carcinoma (MC), and Papillary Carcinoma (PC). Sample descriptions for the BreaKHis dataset are shown in **Table 1**.

Since the input sizes of Inception_V3 and Inception_ResNet_V2 networks used in this paper are both 299 × 299, each of the histopathological images of breast cancer must be transformed into a 299 × 299 image to match the required input size of the network structure. Some image preprocessing methods in the TensorFlow framework were used in the transforming process, including cutting the border box, adjusting image size, and adjusting saturation, etc. In this way, a 3-channel image conforming to the input size of the model was generated, and the pixel values of each channel were normalized to the interval of $[-1, 1]$. In order to ensure the universality of the experimental results in the classification task, the datasets of the four magnification factors were randomly partitioned into training and testing subsets according to the proportion of 7:3.

Classification Analysis

This subsection will discuss our experiments of classifying histopathological images of breast cancer using the deep learning models of Inception_V3 (Szegedy et al., 2016) and Inception_ResNet_V2 (Szegedy et al., 2017) as well as the analyses of our experimental results.

Network Structures for Classification

The Inception_V3 (Szegedy et al., 2016) and Inception_ResNet_V2 (Szegedy et al., 2017) networks, proposed by Szegedy et al. in 2016 and 2017, respectively, were adopted in our experiments. It was demonstrated in the ILSVRC competition that Inception_ResNet_V2 could defeat the Inception_V3 network when applied to big data. An important difference between the Inception_V3 and Inception_ResNet_V2 networks is that the latter is equipped with residual connections. To test whether the experimental results from Inception_ResNet_V2 are superior to those from Inception_V3 on small datasets or not, these two networks are adopted in this paper to perform classification of the histopathological images of breast cancer. The network structures are shown in **Figure 1**.

It can be seen from **Figure 1** that the structures of the two networks are very similar. The first several layers are characteristic transformation via the traditional convolutional layers and the pooling layers, and the middle part is composed of multiple Inception modules stacked together. The results are finally output through the fully-connected layer using the Softmax function. One of the main differences between the Inception_V3 and Inception_ResNet_V2 networks lies in the differing composition of the two networks' Inception modules. To enhance the network's adaptability to different convolution kernels, each Inception module of the Inception_V3 network is composed of filters with different sizes including 1 × 1, 1 × 3, 3 × 1. For the Inception_ResNet_V2 network, to avoid the deterioration of the network gradient that is often associated with an increase in the number of layers, a residual unit is added to each Inception module. Besides using filters of different

TABLE 1 | Image distribution of different subclasses in different magnification factors.

Magnification	Benign				Malignant				Total
	A	F	PT	TA	DC	LC	MC	PC	
40X	114	253	109	149	864	156	205	145	1,995
100X	113	260	121	150	903	170	222	142	2,081
200X	111	264	108	140	896	163	196	135	2,013
400X	106	237	115	130	788	137	169	138	1,820
Total	444	1,014	453	569	3,451	626	792	560	7,909
#Patients	4	10	3	7	38	5	9	6	82

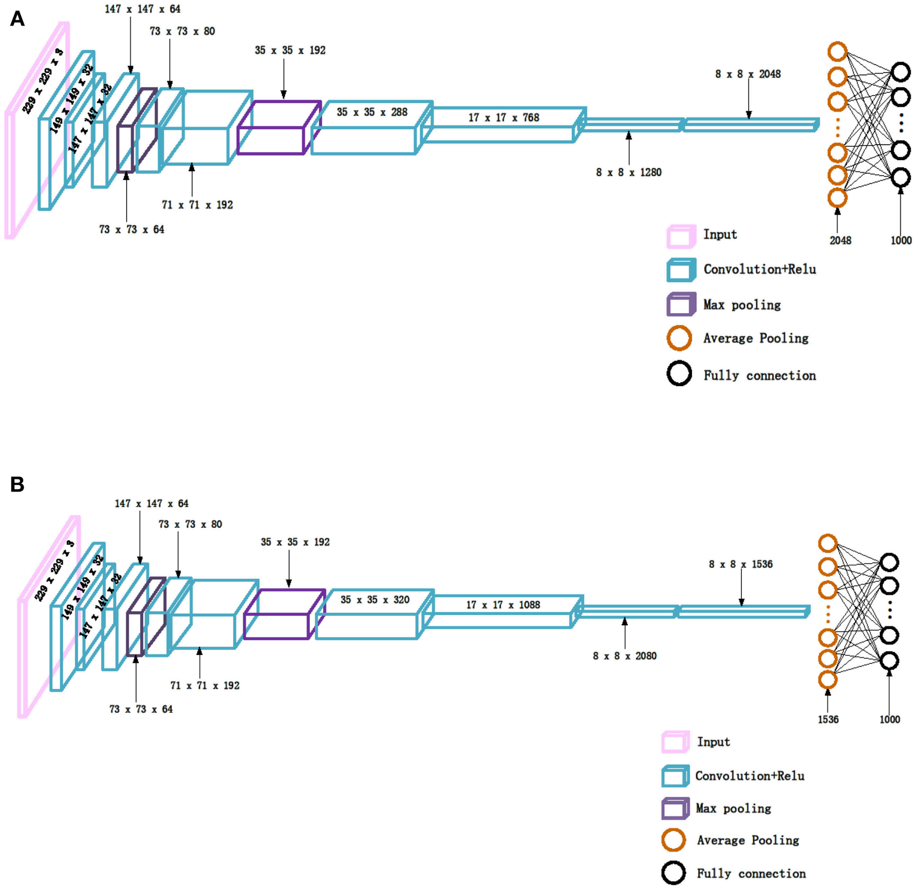


FIGURE 1 | The network structures, **(A)** Inception_V3, **(B)** Inception_ResNet_V2.

sizes in the network, the deterioration caused by increasing layers can also be solved by jumping layers as allowed by the use of residual connections. **Figure 2** displays the differences in the construction of the Inception module with a size of 8 × 8 between Inception_V3 and Inception_ResNet_V2. The other details can be found in the original references (Szegedy et al., 2016, 2017).

Transfer Learning

Transfer learning (Pan and Yang, 2010) emerges from deep learning. It is well-known that it is typically impossible to train a complex deep network from scratch with only a small dataset. Furthermore, there are not any existing principles to design a network structure for a specific task. What we can do is adopt the model and the parameters obtained by other researchers via time-consuming and computationally intensive training on the very large image dataset of ImageNet and use the knowledge it has gained as pre-training for our specific research task. Then, we can retrain the last defined fully-connected layer of the model using only a relatively small amount of data to achieve good results for our target task.

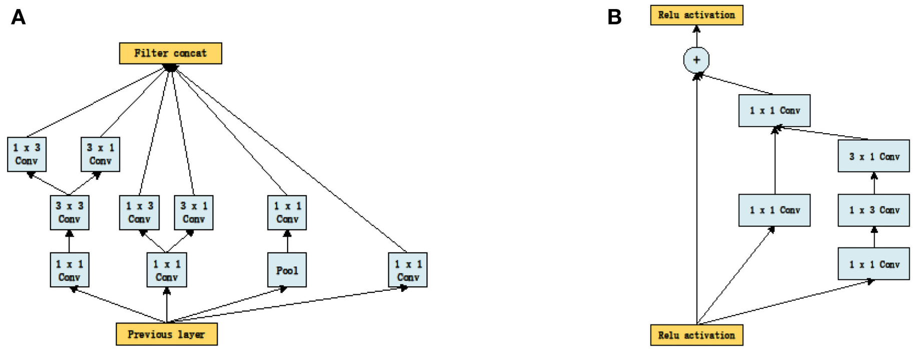


FIGURE 2 | The inception module of size 8 × 8 in two networks, **(A)** Inception_V3, **(B)** Inception_ResNet_V2.

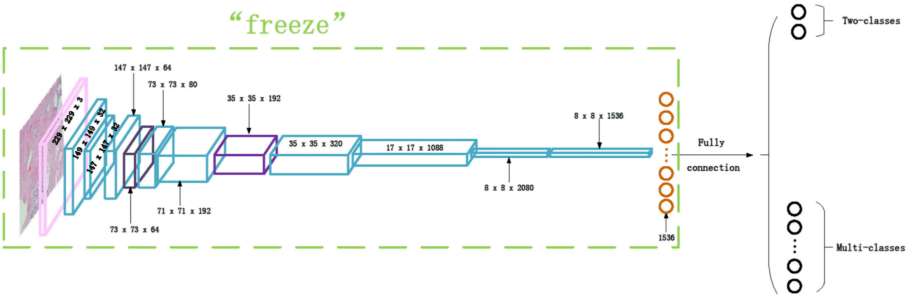


FIGURE 3 | The Inception_ResNet_V2 network structure for transfer learning.

Transfer learning is adopted in this paper to classify the histopathological images of breast cancer using Inception_V3 and Inception_ResNet_V2 networks. We first downloaded the models and parameters of Inception_V3 and Inception_ResNet_V2 networks trained on the ImageNet dataset. The dataset is composed of about 1.2 million training images, 50,000 validation images, and 100,000 testing images. This comprises a total of 1,000 different categories. Then, we froze all of the parameters before the last layer of the networks. We modified the number of neurons of the last fully-connected layer as 2 for binary classification and 8 for multi-class classification. After that, the parameters of the fully-connected layer are trained on the histopathological images of breast cancer. The modified network structure of the Inception_ResNet_V2 network is shown in **Figure 3**. The modified Inception_V3 network structure is similar, so it is omitted.

Our classification process was developed based on the TensorFlow deep learning framework. The Adam (adaptive moment estimation) (Kingma and Ba, 2014) algorithm was used in the training process to perform optimization by iterating through 70 epochs using the histopathological image dataset of breast cancer. The batch_size is set to 32 in the experiments, and the initial learning rate is 0.0002 (Bergstra and Bengio, 2012). Then, the exponential decay method is adopted to reduce the learning rate and ensure that the model moves through iterations quickly at the initial training stage. This also helps to provide more stability at the later stage and makes it easier to obtain the optimal solution. The decay coefficient is set as 0.7 (Bergstra and Bengio, 2012), and the decay speed is set so that the decay occurs every two epochs. The specific decay process is shown in (1), where *decayed_learning_rate* is the current learning rate, *learning_rate* is the initial learning rate, *decay_rate* is the decay coefficient, *global_step* is the current iteration step, and *decay_steps* is the decay speed.

$$decayed\_learning\_rate = learning\_rate \times decay\_rate^{(global\_step/decay\_steps)} \quad (1)$$

Evaluation Criteria for Classification Results

To evaluate the performance of the classification model more accurately and comprehensively, the classification results are evaluated by some popular benchmark metrics, including sensitivity (Se), specificity (Sp), positive predictive value (PPV), diagnostic odds ratio (DOR), F1 measure (F1), area under the receiver operating characteristic curve (AUC), Kappa criteria (Kappa), Macro-F1, Micro-F1, image level test accuracy (ACC_IL), and patient level test accuracy (ACC_PL). The latter two criteria were proposed in (5). The Macro-F1 and Micro-F1 are two variations of F1 for multi-class classification problems. Macro-F1 is the average of F1 for each class. Micro-F1 is defined as F1 but depending on the precision and recall defined by the sum of TP (true positive), FP (false positive), and FN (false negative) for all classes. The definitions of the criteria are shown in Equations (2–9).

$$Se = \frac{TP}{TP + FN} \tag{2}$$

$$Sp = \frac{TN}{TN + FP} \tag{3}$$

$$PPV = \frac{TP}{TP + FP} \tag{4}$$

$$DOR = \frac{TP \times TN}{FP \times FN} \tag{5}$$

$$ACC\_IL = \frac{N_{rec}}{N_{all}} \tag{6}$$

$$ACC\_PL = \frac{\sum Patient\ Score}{Total\ Number\ of\ Patients}, \quad Patient\ Score = \frac{N_{rec}}{N_P} \tag{7}$$

$$F1 = \frac{2 \times precision \times recall}{precision + recall}, \quad recall = \frac{TP}{TP + FN} \tag{8}$$

$$Kappa = \frac{p_0 - p_e}{1 - p_e}, p_0 = \frac{N_{rec}}{N_{all}}, P_e = \frac{\sum N_{true\_i} \times N_{pre\_i}}{N_{all} \times N_{all}} \tag{9}$$

The value of TP in the equations above is the number of images correctly recognized as malignant tumor in the testing subset. FP is the number of images that were incorrectly recognized as malignant tumor in the testing subset. FN is the number of images incorrectly recognized as benign tumor in the testing subset. TN is the number of images correctly recognized as benign tumor in the testing subset. Therefore, Se in (2) defines the ratio of the recognized malignant tumor images to all malignant tumor images in the testing subset. Sp in (3) expresses the ratio of the recognized benign tumor images to all benign tumor images. That is, Se and Sp are the accuracy of the positive and negative class, respectively. PPV in (4) is the ratio of correctly recognized malignant tumor images to all recognized malignant tumor images in the testing subset. In fact, it is the precision in (8). DOR expresses the ratio of the product of TP and TN to the product of FP and FN. It is clear that DOR will become infinity when the related classifier is perfect. It is reported that a diagnosis system is reliable if Se$> =$ 80%, Sp$> =$ 95%, PPV$> =$ 95%, and DOR$> =$ 100 (Ellis, 2010; Colquhoun, 2014). Equation (6) defines image level test accuracy (ACC_IL) by the ratio of N_{rec} (the number of the histopathological images of breast cancer correctly identified in the testing subset), to N_{all} (the total number the histopathological images of breast cancers in the testing subset). Equation (7) defines patient level test accuracy (ACC_PL), that is, the ratio of the sum of patient score to the total number of patients in the testing subset. Here, the patient score is the ratio of N_{rec} to N_P, that is, the ratio of correctly identified images of patient P to all the images of patient P in the testing subset. Equation (8) describes a popular metric known as the harmonic mean of precision and recall. Here, precision is the same as PPV defined as the ratio of correctly recognized malignant tumor images to all recognized malignant tumor images in the testing subset, and recall is the ratio of correctly recognized malignant tumor images to the true number of malignant tumor images in the testing subset. AUC is the area under the ROC curve, which is another widely used metric for evaluating binary classification models. The range of AUC is [0, 1] (Bradley, 1997), with higher values representing better model performance. We calculate AUC in our experiments by calling the roc_auc_score function from the Scikit-learn library that is available as a Python package (sklearn). Equation (9) is the Kappa coefficient, where P_0 is the image level test accuracy defined in (6), and P_e is the ratio of the sum of the product of the number of real images in each category and the predicted number of images in that category to the square of the total samples. The calculation of the Kappa coefficient is based on the confusion matrix. Kappa is used for consistency checking, and its value is in the range of $[-1, 1]$. It can be divided into six groups representing the following consistency levels: $-1\sim0.0$ (poor), $0.0\sim0.20$ (slight), $0.21\sim0.40$ (fair), $0.41\sim0.60$ (moderate), $0.61\sim0.80$ (substantial), and $0.81\sim1$ (almost perfect) (Landis and Koch, 1977).

Clustering Analysis

The classification analysis of histopathological images of breast cancer based on deep convolutional neural networks is introduced in the previous section. However, this type of classification is supervised learning and requires experienced pathologists to examine the histopathological images of breast cancer and assign labels to them that identify the data as coming from patients or normal people. This is very difficult, time-consuming, and expensive work, especially with the increasing number of samples in the dataset. On the contrary, unsupervised learning, specifically clustering, does not need any labels for samples. It only uses the similarities between samples to group them into different clusters, such that the samples in the same cluster are similar to each other and dissimilar to those from other clusters. Therefore, we adopt clustering techniques to study the histopathological images of breast cancer.

Network Structures for Clustering

The Inception_ResNet_V2 network is adopted to extract features for performing clustering analysis of the histopathological images of breast cancer because of its excellent performance when classifying these images using its advantage of extracting features automatically. Each histopathological image of breast cancer can be well-expressed by the extracted features of the 1,536-dimension vector produced by the Inception_ResNet_V2 network before its final classification layer. The extracted feature vectors are used as input to a clustering algorithm in order to perform clustering analysis on the histopathological images of breast cancer.

The very simple and fast, typical clustering algorithm K-means is adopted in this paper to perform this clustering analysis. To determine the proper value of K for the K-means algorithm, the internal criterion metric SSE (Silhouette Score) (Rousseeuw, 1987) is adopted to search for the optimal K. The features extracted by the Inception_ResNet_V2 network for each breast cancer histopathological image are thought of as a representation of the images, and the K-means clustering algorithm is adopted to cluster the breast cancer histopathological images into clusters. Also, in order to get better clustering results and to visualize the clustering results, we constructed a new autoencoder network

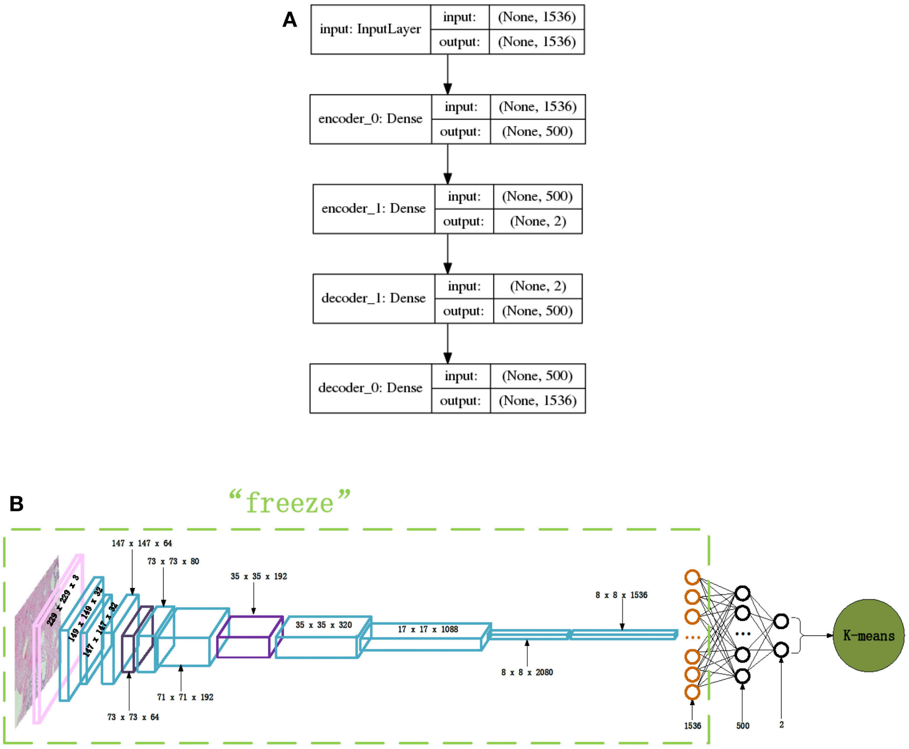


FIGURE 4 | The network structures of our proposed autoencoder and its combination with Inception_ResNet_V2, **(A)** Autoencoder network, **(B)** Inception_ResNet_V2 and autoencoder network.

to map the 1,536-dimension vector to a 2-dimension vector via a non-linear transformation. In this way, the breast cancer histopathological images can be represented in a very low dimensional space. **Figure 4A** displays the autoencoder network we constructed in our experiments. There are 2 encode layers with neuron sizes of 500 and 2, respectively, and there are 2 corresponding decode layers to reconstruct the original input. Using this autoencoder, the 1,536-dimension feature vector extracted by the Inception_ResNet_V2 network for a breast cancer histopathological image will be transformed to 2-dimenision feature vector via training the layers depicted in **Figure 4A**. Then, the 2-dimension feature vector is used as input for K-means which performs the clustering analysis for histopathological images of breast cancer. The entire network is shown in **Figure 4B**.

Evaluation Criteria of Clustering Results

The evaluation criteria of clustering results comprise internal and external metrics. The internal metrics are independent of the external information, so they are always used to find the true number of clusters in a dataset. The external metrics depend on the true pattern of the dataset. Some of the most common external metrics are clustering accuracy (ACC), adjusted rand index (ARI) (Hubert and Arabie, 1985), and adjusted mutual information (AMI) (Vinh et al., 2010).

The internal metric SSE (Silhouette Score) (Rousseeuw, 1987) is used in our experiments. It is first used to find the most proper number of clusters of the histopathological images of breast cancer. Then, after the clustering results have been obtained by K-means, it is used to evaluate the clustering results together with the aforementioned external metrics. Equation (10) gives the Silhouette value of sample ***i***.

$$s\left(\mathbf{i}\right) = \frac{b\left(\mathbf{i}\right) - a\left(\mathbf{i}\right)}{\max\left\{a\left(\mathbf{i}\right), b\left(\mathbf{i}\right)\right\}} \tag{10}$$

Here, $b\left(\mathbf{i}\right)$ is the smallest average distance of sample $\mathbf{i}$ to all samples in any other cluster to which sample $\mathbf{i}$ does not belong. $a\left(\mathbf{i}\right)$ is the mean distance from sample $\mathbf{i}$ to all other samples within the same cluster, and $s\left(\mathbf{i}\right)$ is the Silhouette value of sample $\mathbf{i}$. The average $s\left(\mathbf{i}\right)$ of all samples in a cluster is a measure of how tightly grouped all the samples in the cluster are. Therefore, the average $s\left(\mathbf{i}\right)$ over all samples in an entire dataset is a measure of how appropriately the samples have been clustered; that is what is called the SSE metric.

The external metrics used in this paper are ACC, ARI (Hubert and Arabie, 1985) and AMI (Vinh et al., 2010). It was reported that ARI is one of the best external metrics (Hubert and Arabie, 1985). ARI is defined in (11) and uses the following variables: a (the number of pairs of samples in the same cluster before and after clustering), b (the pairs of samples in the same cluster while partitioned into different clusters by the clustering algorithm), c (the pairs of samples that are from different clusters but are grouped into the same cluster incorrectly by the clustering algorithm), and d (the number of pairs of samples from different clusters that are still in different clusters after clustering). The AMI is defined in (12), where U is the original partition and V is the clustering of a clustering algorithm. Here, $MI\left(U,V\right)$ denotes the mutual information between two partitions U and V, and $E\left\{MI\left(U,V\right)\right\}$ represents the expected mutual information between the original partition U and the clustering V. $H\left(U\right), H\left(V\right)$ are the entropy of the original partition U and the clustering V, respectively. AMI is a variation of mutual information and can be used to compare the clustering V of a clustering algorithm and the true pattern U of the dataset. It corrects the effect of agreement solely due to chance between the clustering and the original pattern. This is similar to the way that the adjusted Rand index corrects the Rand index.

$$ARI = \frac{2(ad - bc)}{(a+b)(b+d)+(a+c)(c+d)} \tag{11}$$

$$AMI\left(U,V\right) = \frac{MI\left(U,V\right) - E\left\{MI\left(U,V\right)\right\}}{\max\left\{H\left(U\right),H\left(V\right)\right\} - E\left\{MI\left(U,V\right)\right\}} \tag{12}$$

We calculate the criteria listed above in our experiments by calling functions embedded in the sklearn library (available as a Python package), such as silhouette_score (SSE), linear_assignment (ACC), adjusted_rand_score (ARI), and adjusted_mutual _info_score (AMI).

EXPERIMENTAL RESULTS AND ANALYSIS

The section will present our classification and clustering experimental results on the 7,909 histopathological images of breast cancer from the BreaKHis dataset and provide some analyses and discussions of the results.

Classification Results

This subsection will present and discuss all of the classification results of histopathological images of breast cancer from BreaKHis dataset provided by Spanhol (Spanhol et al., 2016a). The experimental results include those conducted on the raw dataset and on the augmented dataset. In addition to this, we provide a comparison between our results and the results produced by other researchers.

Experiments on the Raw Dataset

We used the Inception_V3 and Inception_ResNet_V2 networks to perform binary classification of histopathological images of breast cancer into benign and malignant tumors via transfer learning. **Table 2**'s upper part gives the experimental results using Inception_V3 and Inception_ResNet_V2 networks to perform binary classification on the histopathological images of breast cancer in terms of Se, Sp, PPV, DOR, ACC_IL, ACC_PL, F1, AUC and Kappa. In the table, INV3 is the abbreviation for the Inception_V3 network, and IRV2 is the abbreviation for the Inception_ResNet_V2 network.

According to the description of the histopathological image dataset of breast cancer, the benign and malignant tumors can be classified into four different subclasses, respectively. So, there are 8 subclasses in total, including 4 benign tumors (A, F, PT, and TA) and 4 malignant tumors (DC, LC, MC, and PC). The available studies for the histopathological images of breast cancer only focus on binary classification of the images. However, multi-class classification is more significant than binary classification for providing accurate treatment and prognosis for breast cancer patients. Therefore, we did a multi-class classification diagnosis study on the histopathological images of breast cancer by using Inception_V3 and Inception_ResNet_V2 with transfer learning techniques. The experimental results of our multi-class classification of histopathological images of breast cancer are shown in the bottom half in **Table 2** in terms of ACC_IL, ACC_PL, Macro-F1, Micro-F1 and Kappa.

The experimental results in **Table 2** show that the Inception_ResNet_V2 network can get better results in all evaluation metrics compared to the Inception_V3 network, regardless of binary or multi-class classification (which is indicated by the red underline). One reason for this is that residual connections are added to the Inception_ResNet_V2 network, which avoids the vanishing gradient problem typically caused by increasing the number of layers in a network. This also improves the network performance and allows it to extract more informative features from images than Incepiton_V3 can.

Furthermore, the experimental results show that all metrics on the 40X dataset are better than those on the other datasets with any other magnification factors, which is shown in black font. These results are in agreement with those reported in (5). The reason for this should be the 40X dataset containing more significant characteristics of breast cancer.

The experimental results in **Table 2** show that Se>95%, Sp>90%, PPV>95%, and DOR>100 on each dataset regardless of magnification factor and network structure (Inception_V3 or Inception_ResNet_V2). The results from the Inception_ResNet_V2 network show that Se>98%, Sp>92%, PPV>96%, and DOR>100, especially on the 40X dataset where Se >98%, Sp>96%, PPV>98%, and DOR>100. Considering research which suggests that a diagnosis system is reliable when Se> = 80%, Sp> = 95%, PPV> = 95%, and DOR> = 100 (Ellis, 2010; Colquhoun, 2014), we can say that our breast cancer diagnosis system based on the 40X dataset and the Inception_ResNet_V2 network is very reliable. The diagnosis system based on the Incepiton_V3 network is also comparatively reliable.

In addition, the values of AUC and Kappa in **Table 2** tell us that our models are perfect and have obtained almost perfect agreement for binary classification of histopathological images of breast cancer. The values of Kappa in **Table 2** reveal that our models for multi-class classification are also perfect. The models based on the Inception_ResNet_V2 network can get perfect agreement for multi-class classification of breast cancer histopathological images, except when applied to the 400X dataset (which still achieves substantial agreement).

Besides the above analysis, we further verify the power of our approaches for analyzing the breast cancer histopathological images using the p-value of AUC and Kappa. The p-value is a probability that measures the statistical significance of evidence against the null hypothesis. A lower p-value provides stronger

TABLE 2 | Results of binary and multi-class classification using Inception_V3 (INV3) and Inception_ResNet_V2 (IRV2)/%.

Classification	Network	Criteria	Magnification factors			
			40X	100X	200X	400X
Binary	INV3	Se	98.00	98.48	**99.01**	96.41
		Sp	**94.31**	93.46	91.40	90.99
		PPV	**97.41**	96.67	95.88	95.89
		DOR	81,233	92,303	**106,700**	27,105
		ACC_IL	**96.84**	96.76	96.49	94.71
		ACC_PL	**97.74**	94.19	87.21	96.67
		F1	**97.70**	97.56	97.42	96.15
		AUC	**99.47**	99.03	99.29	97.91
		Kappa	92.64	**92.74**	91.95	87.68
	IRV2	Se	98.48	98.90	**99.13**	98.06
		Sp	**96.63**	92.95	92.80	92.10
		PPV	**98.46**	96.45	96.39	96.51
		DOR	**185,774**	118,782	147,138	58,835
		ACC_IL	**97.90**	96.88	96.98	96.98
		ACC_PL	**98.03**	97.07	82.74	88.12
		F1	**98.47**	97.66	97.74	97.28
		AUC	99.57	98.84	**99.61**	98.81
		Kappa	**95.12**	92.96	93.18	91.05
Multi-class	INV3	ACC_IL	**90.28**	85.35	83.99	82.08
		ACC_PL	**90.44**	89.05	80.63	81.08
		Macro-F1	**88.55**	82.59	79.64	77.98
		Micro-F1	**90.28**	85.35	83.99	82.08
		Kappa	**87.37**	80.26	77.91	76.39
	IRV2	ACC_IL	**92.07**	88.06	87.62	84.50
		ACC_PL	**89.11**	88.45	86.07	71.42
		Macro-F1	**90.89**	85.67	84.08	80.13
		Micro-F1	**92.07**	88.06	87.62	84.50
		Kappa	**89.74**	84.03	82.84	79.70

†*For each magnification factor, the underline shows the best result of each evaluation index between the two network structures of INV3 and IRV2. Bold font shows the best result of each evaluation index with respect to the different magnification factors.*

evidence to reject the null hypothesis. Therefore, to determine whether the predictions are due to chance, we calculate the p-values for AUC and Kappa and compare the p-value to the significance level α . It is usually set as $\alpha = 0.05$. We consider both binary and multi-class classification of breast cancer histopathological images with Inception_ResNet_V2 when calculating the p-value for AUC and Kappa. The null hypothesis is "the prediction is a random guess." The p-values for AUC and Kappa are calculated in Equations (13–16) and the pnorm function in R. It should be noted that for multi-class classification, there is only the p-value of Kappa to be calculated.

$$SE_{AUC} = \sqrt{\frac{0.25 + (na + nn - 2)}{na \times nn \times 12}} \tag{13}$$

$$Z_{AUC} = \frac{A - 0.5}{SE_{AUC}} \tag{14}$$

$$SE_{Kappa} = \frac{\sqrt{p_0 \times (1 - p_0)}}{\sqrt{N} \times (1 - p_e)} \tag{15}$$

$$Z_{Kappa} = \frac{Kappa}{SE_{Kappa}} \tag{16}$$

Here, na and nn in (13) are, respectively, the number of abnormal (malignant tumor) and normal (benign tumor) samples (breast cancer histopathological images) in the testing subset. A in (14) is the value of AUC. p_0 and p_e in (15) are the same as those in (9), and N in (15) is the total number of samples. We convert the z value for AUC in (14) and for Kappa in (16) to the corresponding p-value by using the pnorm function in R.

Except for in binary classification, the p-values for AUC are p = 6.88e-85 (40X), p = 2.24e-89 (100X), p = 3.73e-89 (200X), and p = 9.20e-75 (400X). P-values for Kappa are all 0.0, regardless

TABLE 3 | The augmented image distribution of different subclasses in different magnification factors.

Magnification	Benign				Malignant				Total
	A	F	PT	TA	DC	LC	MC	PC	
40X	798	759	763	894	864	936	1,025	870	6,909
100X	791	780	847	900	903	1,020	1,110	852	7,203
200X	777	792	756	840	896	978	980	810	6,829
400X	742	711	805	780	788	822	845	828	6,321
Total	3,108	3,042	3,171	3,414	3,451	3,756	3,960	3,360	27,262
#Patients	4	10	3	7	38	5	9	6	82

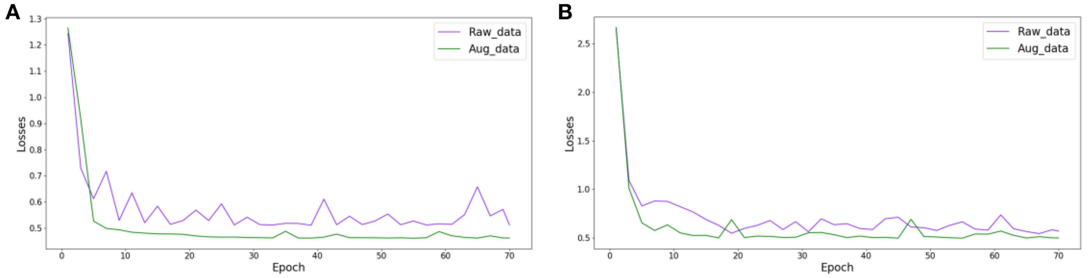


FIGURE 5 | The change in the loss function during the training of Inception_ResNet_V2 on raw and augmented data with 40 factor magnification, **(A)** binary classification, **(B)** multi-class classification.

of binary or multi-class classification. All the *p*-values for AUC and Kappa are much <0.05. This means that we can reject the null hypothesis (that the predictive result is a random guess) and accept that our prediction is statistically significant and not random. This holds true for both our binary and multi-class image classification results.

Experiments on the Augmented Dataset

Comparing the results in **Table 2** for binary and multi-class classification, we can see that the performance of multi-class classification is worse than that of the binary classification. So, we output the confusion matrix of multi-class classification for further analysis. The confusion matrix can be found in the **Supplementary Material**. From observing this confusion matrix, we can see that many benign tumors are incorrectly classified as malignant tumors. This causes a high false positive rate. For example, some samples from F are erroneously recognized as being from DC. Also, the different subclasses in the same class are often misclassified, such as samples from LC being recognized as samples from DC. One reason leading to the poor classification results for multi-class classification is the imbalance in sample distribution. This makes the extracted features unable to thoroughly represent the subclasses with fewer samples. As a result, the samples from the subclass with fewer samples are erroneously classified into the categories with more samples.

To avoid the high false positive rate in multi-class classification, we expanded the original samples of the dataset to suppress the influence that sample imbalance has on the experimental results. For each magnification factor dataset, we chose the DC subclass as the baseline, and amplified each of the remaining subclasses by turning images up and down, left and right, and using counterclockwise rotation of 90°and 180°. After doing this, the sample number of each subclass was approximately the same. The extended datasets are shown in **Table 3**.

We randomly partitioned the extended datasets into training and testing subsets in a 7:3 ratio as we did with the original datasets. Then, we used transfer learning to retrain the Inception_ResNet_V2 network to perform effective diagnosis of breast cancer based on histopathological images of breast cancer. Here, we only retrained the Inception_ResNet_V2 network because it performed better than the Incepiton_V3 network on the raw datasets. To compare the differences in the loss function on the original datasets and on the expansion datasets during the training process, we plotted the value of the loss function changing with the number of epochs in the raw and extended datasets. Here, we only compared the loss function from the Inception_ResNet_V2 network on the 40X dataset in order to observe the changing trend of the loss function. The trends of the other magnification factor datasets are similar. **Figure 5** compared the loss function of the Inception_ResNet_V2 network on the raw and extended datasets, respectively, for binary and multi-class classification of histopathological images of breast cancer. **Table 4** shows the experimental results on the original and expanded datasets for binary and multi-class classification, respectively. The deep learning parameters for both binary and multi-class classification remain the same.

TABLE 4 | Results of binary and multi-class classification on raw and augmented data using Inception_ResNet_V2/%.

Classification	Datasets	Criteria	Magnification factors			
			40X	100X	200X	400X
Binary	Raw_data	Se	98.48	98.90	**99.13**	98.06
		Sp	**96.63**	92.95	92.80	92.10
		PPV	**98.46**	96.45	96.39	96.51
		DOR	**185,774**	118,782	147,138	58,835
		ACC_IL	**97.90**	96.88	96.98	96.98
		ACC_PL	**98.03**	97.07	82.74	88.12
		F1	**98.47**	97.66	97.74	97.28
		AUC	99.57	98.84	**99.61**	98.81
		Kappa	**95.12**	92.96	93.18	91.05
	Aug_data	Se	**99.95**	99.45	99.65	98.88
		Sp	**99.61**	99.26	99.18	99.34
		PPV	**99.66**	99.39	99.31	99.42
		DOR	**56122,884**	2440,736	3427,114	1342,245
		ACC_IL	**99.79**	99.37	99.43	99.10
		ACC_PL	99.93	99.96	**100.0**	99.90
		F1	**99.81**	99.42	99.48	99.15
		AUC	**100.0**	99.99	99.95	99.97
		Kappa	**99.59**	98.72	98.86	98.19
Multi-class	Raw_data	ACC_IL	**92.07**	88.06	87.62	84.50
		ACC_PL	**89.11**	88.45	86.07	71.42
		Macro-F1	**90.89**	85.67	84.08	80.13
		Micro-F1	**92.07**	88.06	87.62	84.50
		Kappa	**89.74**	84.03	82.84	79.70
	Aug_data	ACC_IL	**97.63**	97.00	96.89	97.49
		ACC_PL	**98.42**	98.07	97.85	97.40
		Macro-F1	**97.68**	97.06	97.02	97.48
		Micro-F1	**97.63**	97.00	96.89	97.49
		Kappa	**97.28**	96.55	96.44	97.13

[†] *The underline shows the best result of each metric between the two network structures being compared (INV3 and IRV2). The bold font shows the best result of each metric for each magnification level.*

The results in **Figure 5** show that the value of the loss function decreases much faster and more smoothly converges to a much smaller value on the extended datasets than on the raw datasets. This is true for both experiments on binary and multi-class classification of histopathological images of breast cancer.

The experimental results in **Table 4** show that the experiments on extended datasets have produced much better results than those performed on the raw datasets. This is reflected by the data marked with red underlines, especially the results of multi-class classification on the expanded datasets. These results are a significant improvement compared to those from the original datasets. In addition, the experimental results in **Table 4** tell us that the evaluation metrics of experimental results on 40X datasets are much better than those on any other datasets with different magnification factors, which can also be seen from the values with black fonts in **Table 4**. The results further demonstrate that the 40X dataset should contain more significant characteristics of breast cancer.

The experimental results in **Table 4** for binary classification show that Se>98%, Sp>92%, PPV>96%, and DOR>100 on each dataset regardless of magnification factor or the effects of augmentation (raw or augmented). This is especially true for the results on the augmented datasets where Se>98%, Sp>99%, PPV>99%, and DOR>100. This tells us that the breast cancer diagnosis system based on the augmented dataset and the Inception_ResNet_V2 network is very reliable. Compared to the results in **Table 2**, we can say that augmenting raw imbalanced breast cancer histopathological image datasets can greatly improve the reliability of the diagnosis system.

In addition, the values of AUC in **Table 4** show that our models are excellent. One even achieved the maximum value of AUC (1.0) on the augmented 40X dataset. The values of Kappa in **Table 4** show that our models have obtained perfect agreement for binary classification of histopathological images of breast cancer. The values of Kappa in **Table 4** show that our models are perfect when applied to augmented datasets for multi-class classification.

Furthermore, we calculated the *p*-values for AUC and Kappa on all augmented datasets for binary and multi-class classification. The *p*-values for AUC and Kappa are both 0.0,

TABLE 5 | Binary and multi-class classification comparison between our experimental results and the ones available from other studies /%.

Classification	Criteria	Methods	Magnification factors			
			40X	**100X**	**200X**	**400X**
Binary	ACC_IL	AlexNet_Raw(25)	85.6 ± 4.8	83.5± 3.9	83.1± 1.9	80.8± 3.0
		CSDCNN_Raw(29)	95.8± 3.1	96.9± 1.9	96.7± 2.0	94.9± 2.8
		INV3_Raw	96.84	96.76	96.49	94.71
		IRV2_Raw	97.90	96.88	96.98	96.21
		IRV2_Aug	**99.79**	**99.37**	**99.43**	**99.10**
	ACC_PL	PFTAS+QDA_Raw(5)	83.8± 4.1	82.1± 4.9	84.2± 4.1	82.0± 5.9
		PFTAS+SVM_Raw(5)	81.6± 3.0	79.9± 5.4	85.1± 3.1	82.3± 3.8
		AlexNet_Raw(25)	90.0± 6.7	88.4± 4.8	84.6± 4.2	86.1± 6.2
		CSDCNN_Raw(29)	97.1± 1.5	95.7± 2.8	96.5± 2.1	95.7± 2.2
		INV3_Raw	97.74	94.19	87.23	96.67
		IRV2_Raw	98.03	97.07	82.74	88.12
		IRV2_Aug	**99.93**	**99.96**	**100.0**	**99.90**
Multi-class	ACC_IL	LeNet_Raw(29)	40.1± 7.1	37.5± 6.7	40.1± 3.4	38.2± 5.9
		LeNet_Aug(29)	46.4± 4.5	47.3± 4.9	46.5± 5.6	45.2± 9.1
		AlexNet_Raw(29)	70.1± 7.4	68.1± 7.6	67.6± 4.8	67.3± 3.4
		AlexNet_Aug(29)	86.4± 3.1	75.8± 5.4	72.6± 4.8	84.6± 3.6
		CSDCNN_Raw(29)	89.4± 5.4	90.8± 2.5	88.6± 4.7	87.6± 4.1
		CSDCNN_Aug(29)	92.8± 2.1	93.9± 1.9	93.7± 2.2	92.9± 1.8
		INV3_Raw	90.28	85.35	83.99	82.08
		IRV2_Raw	92.07	88.06	87.62	84.50
		IRV2_Aug	**97.63**	**97.00**	**96.89**	**97.49**
	ACC_PL	LeNet_Raw(29)	38.1± 9.3	37.5± 3.4	38.5± 4.3	37.2± 3.6
		LeNet_Aug(29)	48.2± 4.5	47.6± 7.5	45.5± 3.2	45.2± 8.2
		AlexNet_Raw(29)	70.4± 6.2	68.7± 5.3	66.4± 4.3	67.2± 5.6
		AlexNet_Aug(29)	74.6± 7.1	73.8± 4.5	76.4± 7.4	79.2± 7.6
		CSDCNN_Raw(29)	88.3± 3.4	89.8± 4.7	87.6± 6.4	87.0± 5.2
		CSDCNN_Aug(29)	94.1± 2.1	93.2± 1.4	94.7± 3.6	93.5± 2.7
		INV3_Raw	90.44	89.05	80.63	81.08
		IRV2_Raw	89.11	88.45	86.07	71.42
		IRV2_Aug	**98.42**	**98.07**	**97.85**	**97.40**

[†] *Bold fonts represent the best results among compared approaches with the same classifier.*

which is much <0.05. This fact tells us that we can reject the null hypothesis (that the prediction result is a random guess), and accept the fact that our prediction is statistically significant and not random.

Experimental Comparisons

This subsection will compare the experimental results of classifying histopathological images of breast cancer using the Inception_V3 and Inception_ResNet_V2 networks in addition to a selection of methods from the available studies carried by other research teams. The experimental results will be compared in terms of ACC_IL and ACC_PL, because the available studies only used these two evaluation criteria. The binary and the multi-class classification experimental results are displayed in **Table 5**. Here, INV3_Raw denotes the results obtained by using Inception_V3 on original dataset. IRV2_Raw and IRV2_Aug represent the results produced by Inception_ResNet_V2 on the original and extended datasets, respectively. The bold fonts denote the best results.

The experimental results in **Table 5** tell us that both the evaluation criteria of ACC_IL and ACC_PL applied to the results obtained from the Inception_ResNet_V2 network have the best value among all of the available studies we found in the literature concerning the classification of histopathological images of breast cancer on the expanded datasets for both binary and multi-class classification. The results on the raw datasets produced by the Inception_ResNet_V2 network are better than those produced by other networks. Therefore, the deep learning network of Inception_ResNet_V2 with residual connections is very suitable for classifying the histopathological images of breast cancer. Also, using the expanded histopathological image datasets of breast cancer can obtain better classification and diagnosis results.

To judge whether or not our approaches are statistically significant, we adopted the Friedman's test (Borg et al., 2013) to discover the significant difference between the compared algorithms. If a significant difference has been detected by

TABLE 6 | Results of Friedman's test between our approaches and the compared algorithms at$\alpha = 0.05$.

	Binary classification			Multi-class classification		
	χ^2	*df*	*p*	χ^2	*df*	*p*
ACC_IL	14.6	4	0.0056	30.6667	8	0.0002
ACC_PL	18.1071	6	0.0060	30.8667	8	0.00015

Friedman's test, then the multiple comparison test is used as a *post hoc* test to detect the significant difference between pairs of the compared algorithms. Friedman's test is considered preferable for comparing algorithms over several datasets without any normal distribution assumption (Borg et al., 2013). We conducted Friedman's test at $\alpha = 0.05$ using the results of algorithms on all datasets in terms of ACC_IL and ACC_PL for binary and multi-class classification shown in **Table 5**. The Friedman's test results are shown in **Table 6**. Here, χ^2 is chi-square, *df* is the degree of freedom, and *p* is *p*-value.

The Friedman's test results in **Table 6** tell us that there is a strong significant difference between our approaches and the compared algorithms because any *p* in **Table 6** supports $p \prec 0.05$. Therefore, we conduct a multiple comparison test between each pair of algorithms at the confidence level of 0.95 and show these statistical test results in **Table 7**. The mean rank difference between algorithms is shown in the upper triangle of the table. The statistical significance between pairs of algorithms is displayed in the lower triangle using "*."

The multiple comparison tests in **Table 7** reveal that our breast cancer diagnosis model which uses Inception_ResNet_V2 on the augmented dataset is very powerful. It offers a statistically significant improvement compared to the results from available references that we can find.

This subsection will further compare the experimental results of Inception_ResNet_V2 on histopathological images of breast cancer to those of SVM and 1-NN classifiers with the 1,536-dimension features extracted by the Inception_ResNet_V2 network. Also, it will compare the experimental results of the SVM and 1-NN classifiers with features extracted by other networks.

The experimental results of binary classification of histopathological images of breast cancer with features extracted by Inception_ResNet_V2 are shown in **Table S1** in terms of Se, Sp, PPV, DOR, ACC_IL, ACC_PL, F1, AUC and Kappa. **Table S2** shows the experimental results of multi-class classification of histopathological images of breast cancer with features extracted by Inception_ResNet_V2 in terms of ACC_IL, ACC_PL, Macro-F1, Micro-F1, and Kappa. **Table 8** compared the studies in (5) and ours in terms of ACC_PL, the only evaluation criterion used in (5), when the experimental results are all from SVM and 1-NN classifiers. The differences between our methods and those in (5) are the features. We adopted the Inception_ResNet_V2 network to extract features of histopathological images of breast cancer while those in (5) used other networks to extract features.

The results in the tables in the **Supplementary Material** show that each classifier gets its best experimental results on the extended datasets of histopathological images of breast cancer, regardless of using binary or multi-class classification. The experimental results of the Inception_ResNet_V2 network on the expanded datasets of histopathological images of breast cancer are the best ones among the results from all of the listed classifiers in the tables in the **Supplementary Material**. The experimental results of the SVM and 1-NN classifiers are not better than that of the Softmax classifier, even though the features are extracted by the Inception_ResNet_V2 network. Therefore, it is very appropriate to use the Inception_ResNet_V2 network to classify histopathological images of breast cancer.

The results in **Table 8** reveal that even when using the same classifiers, such as SVM or 1-NN, the experimental results are different. The results based on the extracted features from the Inception_ResNet_V2 network are much better than those in (5) based on the features extracted by other networks. The best results were also obtained using the extended datasets. This analysis further demonstrates that the deep learning network Inception_ResNet_V2 has a powerful ability to extract informative features automatically.

Clustering Results

This subsection will describe the great advantages of Inception_ResNet_V2 network when it is used for automatically extracting informative features from histopathological images of breast cancer. The 1,536-dimension features are extracted by using Inception_ResNet_V2 to process histopathological images of breast cancer, and the K-means clustering algorithm is adopted to group these images into proper clusters. In addition, a new AE (Autoencoder) network with a shape of [1536, 500, 2] is constructed to perform a non-linear transformation to the 1,536-dimension feature vectors produced by Inception_ResNet_V2. In this way, the 2-dimension features of the histopathological images of breast cancer can be obtained for K-means in low dimensional space. Here, IRV2+Kmeans represents the clustering results of K-means with the features extracted by Inception_ResNet_V2, while IRV2+AE+Kmeans represents the clustering results of K-means based on the features transformed by our proposed AE using the features extracted by Inception_ResNet_V2.

Experiments to Find the Number of Clusters in the Dataset

To find the proper K for K-means, we adopt the internal criterion SSE (Silhouette Score) to search for it. The SSE index combines

TABLE 7 | Paired rank comparison of algorithms in ACC_IL and All_PL for binary and multi-class classification.

ACC_IL for binary	IRV2_Aug	IRV2_Raw	INV3_Raw	CSDCNN_Raw(29)	AlexNet_Raw(25)
IRV2_Aug		1.25	2.75	2.0	4.0
IRV2_Raw			1.5	0.75	2.75
INV3_Raw				−0.75	1.25
CSDCNN_Raw(29)					2.0
AlexNet_Raw(25)	*				

ACC_PL for binary	IRV2 _Aug	IRV2 _Raw	INV3 _Raw	CSDCNN _Raw(29)	AlexNet _Raw(25)	PFTAS+SVM _Raw(5)	PFTAS+QDA _Raw(5)
IRV2_Aug		2.75	2.0	2.0	4.0	5.0	5.25
IRV2_Raw			−0.75	−0.75	1.25	2.25	2.5
INV3_Raw				0.0	2.0	3.0	3.25
CSDCNN_Raw(29)					2.0	3.0	3.25
AlexNet_Raw(25)						1.0	1.25
PFTAS+SVM_Raw(5)	*						0.25
PFTAS+QDA_Raw(5)	*						

ACC_IL for multi-class	IRV2 _Aug	IRV2 _Raw	INV3 _Raw	CSDCNN _Aug(29)	CSDCNN _Raw(29)	AlexNet _Aug(29)	AlexNet _Raw(29)	LetNet _Aug(29)	LeNet _Raw(29)
IRV2_Aug		3.0	4.0	1.0	2.5	4.5	6.0	7.0	8.0
IRV2_Raw			1.0	−2.0	−0.5	1.5	3.0	4.0	5.0
INV3_Raw				−3.0	−1.5	0.5	2.0	3.0	4.0
CSDCNN_Aug(29)					1.5	3.5	5.0	6.0	7.0
CSDCNN_Raw(29)						2.0	3.5	4.5	5.5
AlexNet_Aug(29)							1.5	2.5	3.5
AlexNet_Raw(29)								1.0	2.0
LeNet_Aug(29)	*								1.0
LeNet_Raw(29)	*								

ACC_PL for multi-class	IRV2 _Aug	IRV2 _Raw	INV3 _Raw	CSDCNN _Aug(29)	CSDCNN _Raw(29)	AlexNet _Aug(29)	AlexNet _Raw(29)	LetNet _Aug(29)	LeNet _Raw(29)
IRV2_Aug		3.75	3.0	1.0	2.5	4.75	6.0	7.0	8.0
IRV2_Raw			−0.75	−2.75	−1.25	1.0	2.25	3.25	4.25
INV3_Raw				−2.0	−0.5	1.75	3.0	4.0	5.0
CSDCNN_Aug(29)					1.5	3.75	5.0	6.0	7.0
CSDCNN_Raw(29)						2.25	3.5	4.5	5.5
AlexNet_Aug(29)							1.25	2.25	3.25
AlexNet_Raw(29)								1.0	2.0
LeNet_Aug(29)	*								1.0
LeNet_Raw(29)	*								

[†] *The upper triangle shows the difference between algorithms. The lower triangle shows pairs with statistical significance. Asterisks indicate significant difference between the pairs of algorithms in the table.*

the degree of condensation and separation and can be used in cases without any label information. The interval of SSE is [−1, 1]. Higher SSE values are associated with samples belonging to the same cluster being closer together and samples belonging to different groups being farther apart. SSE values closer to 1 indicate better clustering.

The SSE value of clustering of the histopathological images of breast cancer is variable with the number of clusters. **Figure 6** plots the curves of SSE with the number of clusters on the 40X original dataset of histopathological images of breast cancer. The SSE curves of other magnification factor datasets are similar to those in **Figure 6**.

TABLE 8 | Comparison between different networks extracting features for binary classification/%.

Criteria	Methods	Magnification factors			
		40X	100X	200X	400X
ACC_PL	CLBP+SVM_Raw(5)	77.4± 3.8	76.4± 4.5	70.2± 3.6	72.8± 4.9
	GLCM+SVM_Raw(5)	74.0± 1.3	78.6± 2.6	81.9± 4.9	81.1± 3.2
	LBP+SVM_Raw(5)	74.2± 5.0	73.2± 3.5	71.3± 4.0	73.1± 5.7
	LPQ+SVM_Raw(5)	73.7± 5.5	72.8± 5.0	73.0± 6.6	73.7± 5.7
	ORB+SVM_Raw(5)	71.9± 2.3	69.4± 0.4	68.7± 0.8	67.3± 3.1
	PFTAS+SVM_Raw(5)	81.6± 3.0	79.9± 5.4	85.1± 3.1	82.3± 3.8
	IRV2+SVM_Raw	97.93	96.58	97.07	96.62
	IRV2+SVM_Aug	**99.27**	**98.97**	**98.90**	**98.74**
	CLBP+1-NN_Raw(5)	73.6± 2.5	71.0± 2.8	69.4± 1.5	70.1± 1.3
	GLCM+1-NN_Raw(5)	74.7± 1.0	76.8± 2.1	83.4± 3.3	81.7± 3.3
	LBP+1-NN_Raw(5)	75.6± 2.4	73.0± 2.4	72.9± 2.3	71.2± 3.6
	LPQ+1-NN_Raw(5)	72.8± 4.9	71.1± 6.4	74.3± 6.3	71.4± 5.2
	ORB+1-NN_Raw(5)	71.6± 2.0	69.3± 2.0	69.6± 3.0	66.1± 3.5
	PFTAS+1-NN_Raw(5)	80.9± 2.0	80.7± 2.4	81.5± 2.7	79.4± 3.9
	IRV2+1-NN_Raw	97.32	95.91	96.12	95.88
	IRV2+1-NN_Aug	**98.04**	**97.50**	**97.85**	**97.48**

[†] *Bold fonts represent the best results among compared approaches with the same classifier.*

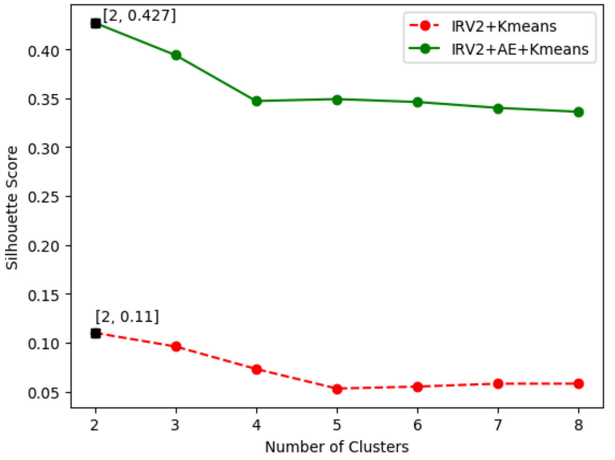


FIGURE 6 | The silhouette score value with different numbers of clusters.

The results in **Figure 6** show the best SSE score was achieved when the number of clusters is 2, regardless of how the features were extracted. This suggests that the histopathological images of breast cancer should be grouped into 2 categories of benign and malignant tumors, which is consistent with the real case. The results in **Figure 6** also reveal that the clustering results of IRV2+AE+Kmeans are better than those from IRV2+Kmeans. This means that the proposed AE network can transform the features extracted by the Inception_ResNet_V2 network into much more informative ones, such that a better clustering of histopathological images of breast cancer can be detected.

Result Evaluation

This subsection will compare the clustering results of IRV2+AE+Kmeans and IRV2+Kmeans in terms of external criteria, including ACC, ARI, AMI, and the internal metric SSE. **Figure 7** displays the clustering results in terms of the aforementioned four evaluation criteria on datasets with different magnification factors.

The experimental results in **Figure 7** reveal the following facts: (1) the clustering results of IRV2+AE+Kmeans are better than those of IRV2+Kmeans in terms of ARI, AMI, SSE, and ACC on each dataset with different magnification factors. This means that our proposed AE network can produce much more abstract and expressive features by encoding the features extracted by the Inception_ResNet_V2 network. (2) The values of ARI, AMI, SSE, and ACC for the same clustering are ascending, regardless of whether or not any transformation has been applied to the features that were extracted by Inception_ResNet_V2. (3) The best clustering accuracy (ACC) with features produced by the Inception_ResNet_V2 network is 59.3% on the 40X dataset, whereas the best ACC with features transformed by the proposed AE network using extracted features from the Inception_ResNet_V2 network is 76.4% on the 200X dataset. In summary, the best ACC scores of IRV2+AE+Kmeans and IRV2+Kmeans are 76.4 and 59.3%, respectively.

CONCLUSIONS AND FUTURE WORK

This paper proposed our methods for the analysis of histopathological images of breast cancer based on the deep convolutional neural networks of Inception_V3 and Inception_ResNet_V2 trained with transfer learning techniques. The aforementioned two networks are pre-trained on the large image dataset of ImageNet. Then, their learned structure and

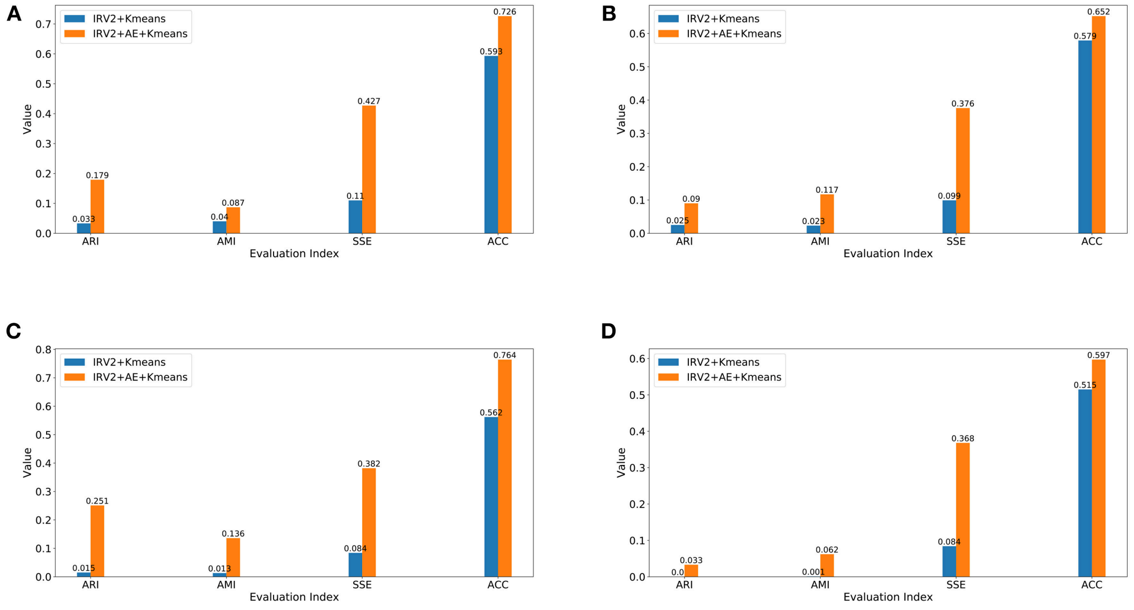


FIGURE 7 | Clustering results in terms of ARI, AMI, SSE, and ACC for datasets with different magnification factors **(A)** 40X, **(B)** 100X, **(C)** 200X, **(D)** 400X.

parameters are frozen. The number of neurons in the last fully-connected layer is set according to our specific task, and the parameters of the fully-connected layer are re-trained. In this way, the model can be used to perform binary or multi-class classification of the histopathological images of breast cancer. We demonstrate that our experimental results are superior to the ones available in other studies that we have found, and that the Inception_ResNet_V2 network is more suitable for performing analysis of the histopathological images of breast cancer than the Inception_V3 network.

Also, our experimental results from the augmented datasets are much better than those from the original datasets. This is especially true when doing multi-class classification with the histopathological images of breast cancer that we used. Our comparison of the experimental results demonstrates that the Inception_ResNet_V2 network is able to extract much more informative features than the other networks we referenced.

The clustering analysis of the histopathological images of breast cancer using the typical clustering algorithm K-means demonstrates that the proper K value for K-means can be found by using the internal criterion SSE. The proposed AE network can detect much more informative, low dimensional features present in histopathological images of breast cancer. Furthermore, the clustering results produced by K-means using features extracted by Inception_ResNet_V2 and transformed by the proposed AE are much better, in terms of ARI, AMI, SSE, and ACC, than the results produced with features extracted only by Inception_ResNet_V2.

All of the work in this paper demonstrates that the deep convolutional neural network Inception_ResNet_V2 has the advantage when it comes to extracting expressive features from histopathological images of breast cancer. The clustering accuracies of histopathological images of breast cancers are not as good as classification accuracies because the latter used label information.

Finding ways that we can improve the clustering accuracy will require further study. In addition to this, finding the number of clusters of histopathological images of breast cancer in both cases of 8 classes and 2 classes is another task that needs to be addressed.

Noise is a prevalent issue in medical imaging and can have a significant effect on results. Some common sources of noise include white patches on slides after deparaffinization, visible patches on tissue after hydrating, and uneven staining. It was reported that batch effects can lead to huge dissimilarities in features extracted from images (Mathews et al., 2016). For the histopathological images used in this paper, it is a fact that the differences of the resolution, contrast and appearance between images from same class are much more apparent than those from different classes. The variance of the fine-grained histopathological images of breast cancer results in difficulties when trying to classify an image as benign, malignant, or another specific category. How we can avoid or reduce the influence on the analysis of histopathological images of breast cancer from these issues will be the focus of our future work.

AUTHOR CONTRIBUTIONS

JX made substantial contributions to the conception and design of the work, drafted the work, and revised it critically for important intellectual content by discussing with CZ, JL, and RL. RL also made substantial contributions to the conception and design of the work. She wrote the code for the algorithms, analyzed the experimental results, and wrote the experimental report. CZ and JL discussed with JX and RL about the technique details, then CZ and JL revised the paper critically for important intellectual content. JX and CZ gave approval for publication of the content. JX and CZ agree to be accountable for all aspects of the work and will ensure that questions related to the accuracy or integrity of any part of the work are appropriately investigated and resolved.

ACKNOWLEDGMENTS

The authors would like to thank Professor Spanhol et al. who provided the database of BreaKHis for us to use, and the other involved researchers for their work to stimulate this research. This work is supported in part by the National Natural Science Foundation of China under Grant No. 61673251. It is also supported by the National Key Research and Development Program of China under Grant No. 2016YFC0901900 and the Fundamental Research Funds for the Central Universities under Grant Nos. GK201701006 and GK201806013. At the same time, it is supported by the Innovation Funds of Graduate Programs at Shaanxi Normal University under Grant Nos. 2015CXS028 and 2016CSY009 as well.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fgene.2019.00080/full#supplementary-material

REFERENCES

Araújo, T., Aresta, G., Castro, E., Rouco, J., Aguiar, P., Eloy, C., et al. (2017). Classification of breast cancer histology images using convolutional neural networks. *PLoS ONE* 12:e0177544. doi: 10.1371/journal.pone.0177544

Asri, H., Mousannif, H., Al Moatassime, H., and Noel, T. (2016). Using machine learning algorithms for breast cancer risk prediction and diagnosis. *Procedia Comput. Sci.* 83, 1064–1069. doi: 10.1016/j.procs.2016.04.224

Aswathy, M., and Jagannath, M. (2017). Detection of breast cancer on digital histopathology images: present status and future possibilities. *Inform. Med. Unlocked* 8, 74–79. doi: 10.1016/j.imu.2016.11.001

Bayramoglu, N., Kannala, J., and Heikkilä, J. (eds) (2016). "Deep learning for magnification independent breast cancer histopathology image classification," in *23rd International Conference on Pattern Recognition (ICPR), 2016.* (Cancun: IEEE).

Bengio, Y., Courville, A., and Vincent, P. (2013). "Representation learning: A review and new perspectives," in *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35, 1798–1828. doi: 10.1109/TPAMI.2013.50

Bergstra, J., and Bengio, Y. (2012). Random search for hyper-parameter optimization. *J. Machine Learn. Res.* 13, 281–305.

Borg, A., Lavesson, N., and Boeva, V. (eds) (2013). *Comparison of Clustering Approaches for Gene Expression Data*. Aalborg: SCAI.

Boyle, P., and Levin, B. (2008). World Cancer report 2008: IARC Press. *International Agency for Research on Cancer*.

Bradley, A. P. (1997). The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recog.* 30, 1145–1159. doi: 10.1016/S0031-3203(96)00142-2

Colquhoun, D. (2014). An investigation of the false discovery rate and the misinterpretation of p-values. *R. Soc. Open Sci.* 1:140216. doi: 10.1098/rsos.140216

Ellis, P. D. (2010). *The Essential Guide to Effect Sizes: Statistical Power, Meta-Analysis, and the Interpretation of Research Results.* New York, NY: Cambridge University Press.

Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., et al. (2017). Dermatologist-level classification of skin cancer with deep neural networks. *Nature* 542, 115–118. doi: 10.1038/nature21056

Filipczuk, P., Fevens, T., Krzyzak, A., and Monczak, R. (2013). Computer-aided breast cancer diagnosis based on the analysis of cytological images of fine needle biopsies. *IEEE Trans. Med. Imaging* 32, 2169–2178. doi: 10.1109/TMI.2013.2275151

George, Y. M., Zayed, H. H., Roushdy, M. I., and Elbagoury, B. M. (2014). Remote computer-aided breast cancer detection and diagnosis system based on cytological images. *IEEE Syst. J.* 8, 949–964. doi: 10.1109/JSYST.2013.2279415

Gulshan, V., Peng, L., Coram, M., Stumpe, M. C., Wu, D., Narayanaswamy, A., et al. (2016). Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. *JAMA* 316, 2402–2410. doi: 10.1001/jama.2016.17216

Han, Z., Wei, B., Zheng, Y., Yin, Y., Li, K., and Li, S. (2017). Breast cancer multi-classification from histopathological images with structured deep learning model. *Sci. Rep.* 7:4172. doi: 10.1038/s41598-017-04075-z

Haralick, R. M., Shanmugam, K., and Dinstein, I. H. (1973). Textural features for image classification. *IEEE Transac. Syst. Man Cybernet.* 3, 610–621. doi: 10.1109/TSMC.1973.4309314

He, K., Zhang, X., Ren, S., and Sun, J. (*eds)* (2016)."Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Las Vegas, NV).

He, K., Zhang, X., Ren, S., and Sun, J. (eds) (2015). "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE international conference on computer vision* (Santiago).

Hinton, G. E., and Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science* 313, 504–507. doi: 10.1126/science.1127647

Hubert, L., and Arabie, P. (1985). Comparing partitions. *J. Classification* 2, 193–218. doi: 10.1007/BF01908075

Kingma, D. P., and Ba, J. (2014). *Adam: a Method for Stochastic Optimization*. arXiv preprint arXiv:14126980.

Kowal, M., Filipczuk, P., Obuchowicz, A., Korbicz, J., and Monczak, R. (2013). Computer-aided diagnosis of breast cancer based on fine needle biopsy microscopic images. *Comput. Biol. Med.* 43, 1563–1572. doi: 10.1016/j.compbiomed.2013.08.003

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (eds) (2012). Imagenet classification with deep convolutional neural networks. *Adv. Neural Inform. Proce. Syst.* 60, 84–90. doi: 10.1145/3065386

Landis, J. R., and Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics* 33, 159–174.

Lowe, D. G. (ed) (1999). "Object recognition from local scale-invariant features." in *The Proceedings of the Seventh IEEE International Conference on Computer Vision*. (Kerkyra: IEEE).

MacQueen, J. (eds) (1967). "Some methods for classification and analysis of multivariate observations," in *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability* (Oakland, CA).

Mathews, A., Simi, I., and Kizhakkethottam, J. J. (2016). Efficient diagnosis of cancer from histopathological images by eliminating batch effects. *Procedia Technol.* 24, 1415–1422. doi: 10.1016/j.protcy.2016.05.165

Moraga-Serrano, P. E. (2018). Global, regional, and national cancer incidence, mortality, years of life lost, years lived with disability, and disability-adjusted life-years for 29 cancer groups, 1990 to 2016: a systematic analysis for the global burden of disease study. *JAMA Oncol*. 4, 1553–1568. doi: 10.1001/jamaoncol.2018.2706

Motlagh, N. H., Jannesary, M., Aboulkheyr, H., Khosravi, P., Elemento, O., Totonchi, M., et al. (2018). *Breast Cancer Histopathological Image Classification: a Deep Learning Approach.* bioRxiv 242818. doi: 10.1101/242818

Nawaz, M., Sewissy, A. A., and Soliman, T. H. A. (2018). Multi-class breast cancer classification using deep learning convolutional neural network. *Int. J. Adv. Comput. Sci. Appl.* 9, 316–322. doi: 10.14569/IJACSA.2018.090645

Pan, S. J., and Yang, Q.,(2010). A survey on transfer learning. *IEEE Transac. Knowledge Data Eng.* 22, 1345–1359. doi: 10.1109/TKDE.2009.191

Rousseeuw, P. J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.* 20, 53–65. doi: 10.1016/0377-0427(87)90125-7

Spanhol, F. A., Oliveira, L. S., Petitjean, C., and Heutte, L. (2016a). A dataset for breast cancer histopathological image classification. *IEEE Transac. Biomed. Eng.* 63, 1455–1462. doi: 10.1109/TBME.2015.2496264

Spanhol, F. A., Oliveira, L. S., Petitjean, C., and Heutte, L. (eds) (2016b). "Breast cancer histopathological image classification using convolutional neural networks," in *2016 International Joint Conference on Neural Networks (IJCNN).* (Vancouver, BC: IEEE).

Stenkvist, B., Westman-Naeser, S., Holmquist, J., Nordin, B., Bengtsson, E., Vegelius, J., et al. (1978). Computerized nuclear morphometry as an objective method for characterizing human cancer cell populations. *Cancer Res.* 38, 4688–4697.

Szegedy, C., Ioffe, S., Vanhoucke, V., and Alemi, A. A. (eds) (2017). *Inception-v4, Inception-Resnet and the Impact of Residual Connections on Learning*. AAAI

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (eds) (2016). "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Las Vegas, NV).

Veta, M., Pluim, J. P., van Diest, P. J., and Viergever, M. A. (2014). Breast cancer histopathology image analysis: a review. *IEEE Transac. Biomed. Eng.* 61, 1400–1411. doi: 10.1109/TBME.2014.2303852

Vinh, N. X., Epps, J., and Bailey, J. (2010). Information theoretic measures for clusterings comparison: variants, properties, normalization and correction for chance. *J. Machine Learn. Res.* 11, 2837–2854.

Xie, J., Hou, Q., Shi, Y., Peng, L., Jing, L., Zhuang, F., et al. (2018). *The Automatic Identification of Butterfly Species. arXiv [preprint]. arXiv:180 306626*.

Zhang, Y., Zhang, B., Coenen, F., and Lu, W. (2013). Breast cancer diagnosis from biopsy images with highly reliable random subspace classifier ensembles. *Machine Vision Appl.* 24, 1405–1420. doi: 10.1007/s00138-012-0459-8

Zhang, Y., Zhang, B., Coenen, F., Xiao, J., and Lu, W. (2014). One-class kernel subspace ensemble for medical image classification. *EURASIP J. Adv. Signal Proces.* 2014:17. doi: 10.1186/1687-6180-2014-17

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

frontiers in Bioengineering and Biotechnology

ORIGINAL RESEARCH
published: 28 March 2019
doi: 10.3389/fbioe.2019.00065

Optimization of a Deep-Learning Method Based on the Classification of Images Generated by Parameterized Deep Snap a Novel Molecular-Image-Input Technique for Quantitative Structure–Activity Relationship (QSAR) Analysis

*Yasunari Matsuzaka and Yoshihiro Uesawa**

Department of Medical Molecular Informatics, Meiji Pharmaceutical University, Tokyo, Japan

OPEN ACCESS

Edited by:
Chaoyang Zhang,
University of Southern Mississippi,
United States

Reviewed by:
Dong Xu,
University of Missouri, United States
Leihong Wu,
National Center for Toxicological
Research (FDA), United States
Sundar Thangapandian,
Engineer Research and Development
Center (ERDC), United States

****Correspondence:***
Yoshihiro Uesawa
uesawa@my-pharm.ac.jp

Specialty section:
This article was submitted to
Bioinformatics and Computational
Biology,
a section of the journal
Frontiers in Bioengineering and
Biotechnology

Received: *28 September 2018*
Accepted: *07 March 2019*
Published: *28 March 2019*

Citation:
Matsuzaka Y and Uesawa Y (2019)
Optimization of a Deep-Learning
Method Based on the Classification of
Images Generated by Parameterized
Deep Snap a Novel
Molecular-Image-Input Technique for
Quantitative Structure–Activity
Relationship (QSAR) Analysis.
Front. Bioeng. Biotechnol. 7:65.
doi: 10.3389/fbioe.2019.00065

Numerous chemical compounds are distributed around the world and may affect the homeostasis of the endocrine system by disrupting the normal functions of hormone receptors. Although the risks associated with these compounds have been evaluated by acute toxicity testing in mammalian models, the chronic toxicity of many chemicals remains due to high cost of the compounds and the testing, etc. However, computational approaches may be promising alternatives and reduce these evaluations. Recently, deep learning (DL) has been shown to be promising prediction models with high accuracy for recognition of images, speech, signals, and videos since it greatly benefits from large datasets. Recently, a novel DL-based technique called DeepSnap was developed to conduct QSAR analysis using three-dimensional images of chemical structures. It can be used to predict the potential toxicity of many different chemicals to various receptors without extraction of descriptors. DeepSnap has been shown to have a very high capacity in tests using Tox21 quantitative qHTP datasets. Numerous parameters must be adjusted to use the DeepSnap method but they have not been optimized. In this study, the effects of these parameters on the performance of the DL prediction model were evaluated in terms of the loss in validation as an indicator for evaluating the performance of the DL using the toxicity information in the Tox21 qHTP database. The relations of the parameters of DeepSnap such as (1) number of molecules per SDF split into (2) zoom factor percentage, (3) atom size for van der waals percentage, (4) bond radius, (5) minimum bond distance, and (6) bond tolerance, with the validation loss following quadratic function curves, which suggests that optimal thresholds exist to attain the best performance with these prediction models. Using the parameter values set with the best performance, the prediction model of chemical compounds for CAR agonist was built using 64 images, at 105° angle, with AUC of 0.791. Thus, based on these parameters, the proposed DeepSnap-DL approach will be highly reliable and beneficial to establish models to assess the risk associated with various chemicals.

Keywords: chemical structure, constitutive androgen receptor, Deep Snap, Tox21, deep learning, QSAR, molecular image

INTRODUCTION

The traditional human-safety assessment of chemical compounds involves repetitive-dosage subacute toxicity testing *in vivo* using animal models. However, the risk remains that such compounds could pose major public health concerns to humans by potentially disrupting normal endocrine functions with various hormone receptors upon long-term exposure (Genuis and Kyrillos, 2017; Heindel et al., 2017; Manibusan and Touart, 2017; Sifakis et al., 2017; Tapia-Orozco et al., 2017; Heindel, 2018; Marty et al., 2018). However, since some molecular mechanisms differ between species and depend on environmental factors, it is often difficult to apply the outcomes of animal testing to predict the effects on human health (Brockmeier et al., 2017; Leist et al., 2017; Fay et al., 2018). Moreover, a large number of chemical substances need to be studied to identify the adverse effects on development, metabolic homeostasis, reproduction, cytotoxicity, etc. (Zhu et al., 2014; Bell et al., 2017; Insel et al., 2017; Juberg et al., 2017; Clark and Steger-Hartmann, 2018; Mortensen et al., 2018). Thus, high-throughput (HTP) assays and economical methods are required (Tollefsen et al., 2014; Chen et al., 2015; Wang et al., 2015; Richard et al., 2016). Alternative computational prediction methods based on *in-silico* experiments are essential for conducting safety evaluations of high-risk chemical substances (Malloy et al., 2017; Lo et al., 2018; Luechtefeld et al., 2018; Zhang et al., 2018). Among these, quantitative structure–activity relationship (QSAR) analysis can predict physiological activity, toxicity, enzymatic reactions, receptor agonist/antagonist activity, environmental fate, etc. (Bloomingdale et al., 2017; Polishchuk, 2017; Halder et al., 2018; Khan and Roy, 2018; Simões et al., 2018). This analysis is conducted based on a formulation of established rules for the relationship between the chemical structure of a compound and its activity and relies on the structural, quantum chemical, and physicochemical features, which are represented as various numerical molecular descriptors (Dougall, 2001; Fang et al., 2003; Roy and Das, 2014; Silva and Trossini, 2014). However, there are limited programs that can precisely evaluate the response patterns of cellular signaling molecules due to various chemical compounds.

These days, machine learning has been applied in extensive toxicological fields, and it is highly effective for risk assessment (Ambe et al., 2018; Banerjee et al., 2018; Luechtefeld et al., 2018; Cipullo et al., 2019). More recently, deep learning (DL), a machine-learning method designed to extract and recognize discriminative information patterns and rules, has been proposed to identify features by several flexible fully-connected layers of a neural network (NN) (Li S. et al., 2017; Qiu et al., 2017; Hu et al., 2018; Li H. et al., 2018; Luechtefeld et al., 2018; Mayr et al., 2018). Until today, support vector machine, random forest, and artificial NN were needed to select a reasonable combination of features (corresponding to chemical structure descriptors in QSAR analysis) manually when learning (feature selection techniques). In many cases, it is extremely difficult to find the optimal solutions, since myriad (Manallack et al., 2010; Talevi et al., 2012; Guimarães et al., 2016; Fang et al., 2017). Therefore, various approximation methods have been developed to obtain an optimal combination for an approximate solution (Yap et al., 2007; Kulkarni et al., 2009). However, since there is no completely trustworthy approximation method, complicated craftsmanship procedures are required to extract effective features in conventional machine learning.

On the other hand, a convolutional neural network (CNN) that constitutes DL has a function of feature expression learning that makes it automatically extract features and unnecessary to manually extract features (Fernandez et al., 2018; Lumini and Nanni, 2018). Unlike the conventional method, which is essential for extraction of a molecular structure descriptor, it is able to identify the most informative features required automatically, which is useful for prediction from the input information of the entire molecule "without supervision" by hierarchically decomposing an image so that the CNN learns to recognize higher-quality features while maintaining their spatial relationships (Ma et al., 2015; Ragoza et al., 2017; Xu et al., 2017; Ghasemi et al., 2018; Liu R. et al., 2018; Peng et al., 2018). These layer structures of the DL consist of input, hidden intermediate, and output layers of a NN, which is an algorithm designed for pattern recognition where information flows and is referred to as a deep neural network (DNN) (LeCun et al., 2015; Mallat, 2016; Suárez-Paniagua and Segura-Bedmar, 2018; Voulodimos et al., 2018). In this DNN, it is possible to directly learn feature quantity contained in a large amount of input data without human intervention at each layer (Azimi et al., 2018). Moreover, it poses a capacity to improve the prediction accuracy for very complicated image recognition by increasing the information transmission and processing ability using a large number of hidden layers and some techniques such as dropout, data augmentation, Rectified Linear Units (ReLUs), and multiple graphics processing units (GPUs) (Rawat and Wang, 2017; Gawehn et al., 2018; Ha et al., 2018; Hussain et al., 2018; Poernomo and Kang, 2018; Qiao et al., 2018; Saha et al., 2018; Sato et al., 2018; Shen et al., 2018; Steven and Han, 2018; Tustison et al., 2018; Vakli et al., 2018; Wang S. H. et al., 2018). Therefore, it is also possible to cope with the deviation and the deformation of the position of input image data for detecting on the edge region

(Krizhevsky et al., 2012). However, since the result depends on the size of the filter, the moving width, and settings such as padding (the process of filling that allocates the end of region with 0 to pad out the number of convolutions of the edge region of the image) (Szegedy et al., 2014; Johnson and Zhang, 2015). In addition, CNNs appropriate combinations of extracted constituent elements and data orderly to the next layer, so it is possible to efficiently learn feature quantities (Szegedy et al., 2014; Cagli et al., 2017).

Studies have reported very high prediction accuracy DL with highly non-linear hierarchical patterns based on large-scale data, especially in the fields of imaging and toxicology (LeCun et al., 2015; Ma et al., 2015; Mayr et al., 2016; Pastur-Romay et al., 2016; Zhang et al., 2017). In addition, some studies have demonstrated the use of DL in QSAR analysis to calculate feature values from molecular structures without human intervention that three steps: (1) model building from labeled data inputs, (2) evaluation and tuning of the model, and (3) training the final model to perform prediction (Bengio et al., 2013; LeCun et al., 2015; Ma et al., 2015; Mayr et al., 2016; Pastur-Romay et al., 2016; Pham et al., 2017; Zhang et al., 2017). However, since for delivering information on the whole molecule sufficiently established most of the cases where DL is applied to QSAR on conventional descriptor calculation at present. Therefore, further work is required to increase prediction accuracy for applications DL for QSAR analysis. First, a systematic and suitable input is required for complicated data such as the three-dimensional (3D) structures of chemical compounds. Moreover, as a result of the insufficient amount of chemical compounds, there is a lack of training data. To address these issues, a novel QSAR model using DL based on 3D molecular images of chemical compounds was previously developed (Uesawa, 2018).

Deep Snap is a procedure of generating an omnidirectional snapshot portraying 3D structures of chemical compounds using a drawing software (Jmol; Hanson, 2016) based on the Structure Data File (SDF) format (**Figure 1**). The 3D information is input into the DL model without calculating structural descriptors. For example, when the 3D molecular structure is rotated in 45° increments on the x-, y-, and z-axes and photographed, a total of 512 images are captured for each molecule and saved in the portable network graphics (PNG) format. This allows for combining digital information regarding the 2D plane location of the atoms with pixel-level data representing the three primary colors (RGB) (**Figure 1**; Uesawa, 2018). Then, these images are used in inputs of the DL model after a resolution of 256 × 256 pixels images of the 3D molecular structure are represented as a ball-and-stick model for each atomic composition with different colors representing different atoms (Uesawa, 2018). We refer to this omnidirectional snapshot capturing procedure for 3D structures of compounds as "Deep Snap."

In the Tox21 data challenge in 2014, a crowd-sourced QSAR competition for chemical risk assessment held by the National Institutes of Health (NIH) in the United States (Tox21 Data Challenge., 2014), approximately 7,000–9,000 different chemical structures depending on the target type. This data was split evenly into training and validation datasets (a 50% of training and a 50% of validation) that were created for the purpose of developing high-performance prediction models for various adverse-outcome pathways (Attene-Ramos et al., 2013; Tox21 Data Challenge., 2014. Recently, using a set of these chemicals (containing a total of 7,320 different molecules with 3,660 reserved for training and 3,660 reserved for validation), the Deep Snap procedure was applied to successfully predict which chemical compounds disrupt the potential of the mitochondrial membrane (MMP), which play pivotal roles in apoptosis, oxidative phosphorylation, calcium homeostasis, and cellular metabolism such as heme, fatty acid, and steroid synthesis (Midzak et al., 2011; Hua et al., 2012; Bolisetty et al., 2013; Shaughnessy et al., 2014; Li A. X. et al., 2017; Liu et al., 2017; Yun et al., 2017; Wang C. et al., 2018). Individual compounds well-known inhibitors for complex between uncouplers (e.g., Carbonyl cyanide-p-trifluoromethoxyphenylhydrazone: FCCP) and particular protein/complex in the transporter chain (rotenone and antimycin A) have been detected in 76 structurally related clusters from the Tox21 10K library (Attene-Ramos et al., 2015; Xia et al., 2018). As potential mitochondrial toxicants, these compounds were found to cause significant reduction of the MMP using an MMP assay in HepG2 cells and rat hepatocytes (Attene-Ramos et al., 2015; Xia et al., 2018). Using transfer learning techniques and an unmodified version of the AlexNet network, the prediction model developed by the Deep Snap-DL method showed area under the ROC curve (AUC) value of 0.921 in the external validation, which included only 647 of the chemical structures employed previously by the Tox 21 Data Challenge 2014 (Uesawa, 2018). At the Tox 21 Data Challenge 2014 competition, the best AUC = 0.95 (Abdelaziz et al., 2016). The prediction performance (AUC = 0.921) by the Deep Snap-DL method is equal to top 10th in the Tox 21 Data Challenge 2014 competition (Tox21 Data Challenge., 2014; Uesawa, 2018. The result suggests that the DL approach based on Deep Snap is suitable for modeling to support toxicological assessments. However, further improvements are required for speed, automation, optimization, and efficiency. Despite the requirement for these improvements, herein, we examine the parameters for Deep Snap and DL to characterize how they affect the DNNs.

MATERIALS AND METHODS

Data

Chemical substance profiles for cellular toxicity were collected from the publicly available Tox21 10K chemical library, 12,500 chemical substances, including pesticides, industrial, food-use, and drugs, procured from commercial sources screened by the Toxicology in the 21st Century (Tox21) program, a multi-agency collaboration between the U.S. Environmental Protection Agency, the National Institute of Environmental Health Sciences, National Toxicology Program, NIH Chemical Genomics Center, National Center for Advancing Translational Sciences, and the US Food and Drug Administration (1) incorporate advances in molecular systems by identifying patterns of chemical compounds-induced biological response, (2) prioritize compounds for more extensive toxicological evaluation, and (3) develop predictive models for biological response in human

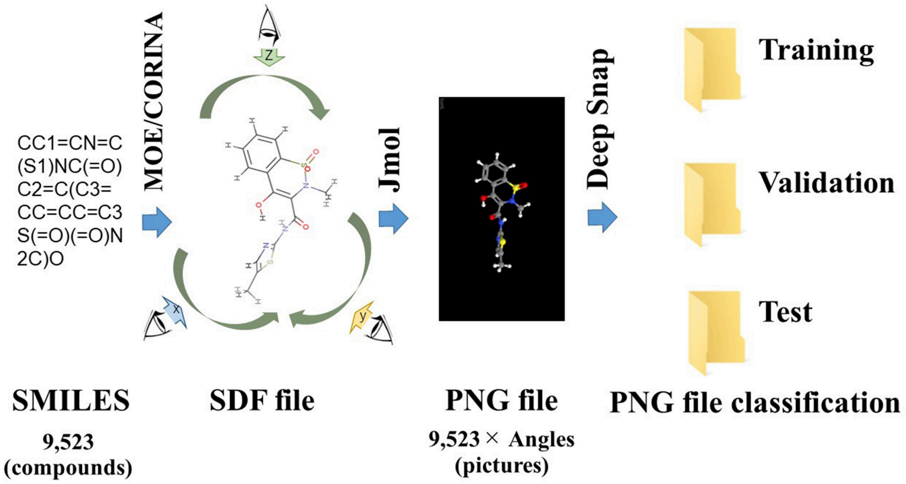


FIGURE 1 | Schematic of the Deep Snap procedure. 9,523 SMILES 3D structures by CORINA Classic software after washing by MOE application, and into SDF file format, and then photograph an arbitrary angle on the x-, y-, and z-axes by Jmol-Deep Snap. The resulted images are saved as PNG files in three datasets (training, validation, and test) in order to input DL.

(NRC., 2007 Collins et al., 2008; Kavlock et al., 2009; Huang et al., 2011, 2014, 2016; Attene-Ramos et al., 2013; Tice et al., 2013; Chen et al., 2015; Hsieh et al., 2015, 2017; Merrick et al., 2015; Huang and Xia, 2017; Sipes et al., 2017). Their structures and the corresponding activities were used to determine agonist of a constitutive androstane receptor (CAR; NR1I3), which is a member of the ligand-activated superfamily of nuclear receptors transcriptionally activated genes predominantly expressed in the liver such as *CYP2B6* and *CYP3A4* involved in not only all phases of drug metabolism, transport, detoxification, and disposition about 50% of the drug metabolization potential in the body but also energy metabolism, tumor progression, cholesterol homeostasis, and glucose metabolism (Qatanani and Moore, 2005; Kobayashi et al., 2015; McMahon et al., 2018).

Deep Snap Procedure: Creation of Molecular Image Files

A total of 9,667 of the chemical structures and the corresponding labeled activity scores were downloaded in the SMILES (Simplified molecular input line entry system) format (Weininger, 1988; Putz and Dudaş, 2013; Achary, 2014; Kumar and Chauhan, 2018) from the PubChem database (AID 1224892) derived from Tox21 10k library, the activity scores defined as the Pubchem_activity_scores (zero and scores between 1 and 100 were represented as inactive and active compounds, respectively, by cell viability and agonist activity screenings of the CAR signaling pathway). Then, by eliminating non-organic compounds, a total of 9,523 of the chemical compounds were selected (**Table 1**; **Supplementary Table 1**). After structure cleaning and standardization (removing salts, counterions, and fragments) by conformational import that is a high-throughput conformer generation method for large numbers of molecules using the MOE application software program (but no treatment of protonation states) (Chen and Foloppe, 2008; Molecular Operating Environment, Chemical Computing Group, Canada) (**Supplementary Table 1**), one 3D chemical structure per compound which have "rotatable torsions" was curated and optimized to generate a single low energy conformation using CORINA Classic software (Molecular Networks GmbH, Nürnberg, Germany, https://www.mn-am.com/products/corina) has been licensed in the past to predict 3D structures for some of the molecules in the main large public databases of small molecules such as PubChem a data-based commercial 3D molecular model builder with high accuracy and high speed for the 3D-structures of organic and metal-organic (also known as organometallic) molecules high coverage for nearly all organics but approximately half of the organometallics (Sadowski et al., 1994; Reitz et al., 2004; Tetko et al., 2005; Renner et al., 2006; Wang et al., 2009; Schwab, 2010; Andronico et al., 2011; Sayers et al., 2018; 3D Structure Generator CORINA Classic., 2019). Finally, these chemical structures were converted to the SDF file format. During the Deep Snap process, when the number of molecules described in the SDF file is large, the power required for the describing. Therefore, in order to improve the depiction speed, it is possible to multiple processes to be executed simultaneously by partitioning of the input data. The size of PNG file is different depending on the number of per SDF file. Moreover, the csv file including annotation data numbers, activity score, and dataset types that was divided randomly into training (4,761 chemicals), validation (2,381 chemicals), and testing (2,381 chemicals) datasets (**Table 1**; **Supplementary Table 1**) was used as the source for labeling each sample. Since the 3D-chemical structures can rotate 360° on each snapshots were captured at a range of fixed increments based on the SDF molecular structure file and the using a novel technique to capture generated images by their description function without human intervention saved as 256 × 256 (pixels resolution) PNG files (RGB) organized by their annotation data numbers (**Figure 1**). In this study, the 3D structure data was preliminarily portrayed as ball-and-stick

TABLE 1 | Number of chemical compounds in train, validation, and test datasets used in optimization of parameter of Deep Snap.

Activity score	Training	Validation	Test	Sum
0: Non-toxic	3,651	1,858	1,878	7,387
1: Toxic	1,110	523	503	2,136
Sum	4,761	2,381	2,381	9,523

structures in four types of increments on the x-, y-, and z-axes: first was (0,0,0), second was (0,0,0), (0,90,0), and (0,0,90), third was (0,0,0), (180,0,0), (0,180,0), and (0,0,180), fourth was (0,0,0), (180,0,0), (0,180,0), (0,0,180), (0,180,180), (180,0,180), (180,180,0), and (180,180,180) included 4 overlapped images automatically and manually obtained from the Deep Snap process, respectively to assess the systematic and suitable input of the 3D structures of chemical compounds and optimization Deep Snap (**Figures 2A–H**). The 3D ball-and-stick model with different colors to different atoms represented by which uses a unique algorithm to calculate surfaces (Jmol, Herráez, 2006; Cammer, 2007; Hanson, 2016; Scalfani et al., 2016; Hanson and Lu, 2017). More detailed technical information is available at the Jmol website[1] As for the depiction process in Deep Snap, it is possible to design a setting cfg file that can specify arbitrary of the Jmol script such as image pixel size, image format (png or jpg), number of molecules per sdf file to split into (MPS), zoom factor (ZF, %), atom size for van der waals (AT, %), bond radius (BR) (mÅ), minimum bond distance (MBD), bond tolerance (BT), etc. Finally, using 64 pictures 105° angle and (MPS:100, ZF:100, AT:23, MBD:0.4, BT:0.8) as permutation test to assess non-specific activity score, they were randomly reassigned based on the activity scores without changing training, validation, and test datasets. Using a total of 10 different datasets, the prediction models were constructed by Deep Snap-DL method with the parameter values for the best performance optimized in this study eight pictures at 180° angle.

Machine-Learning Models Based on DL

All the two-dimensional (2D) images contained digitized information data about plane configuration and the corresponded to the type of atom for the chemical structure produced by Deep Snap were resized by DIGITS version 4.0.0 software to a fixed resolution of 256 × 256 pixels and input into DL model to build the prediction models, which were trained based on the activity scores of chemical compounds and the corresponding 2D chemical-structure images. In this study, the total number of training epochs was 30, snapshot interval in epochs 1, validation interval in epochs 1, random seed 1, solver type stochastic gradient descent, base learning rate 0.01. Training, testing, and validation were performed using the dataset described in **Table 1** and **Supplementary Table 2**. Finally, the performance of the prediction model was evaluated using one test dataset not used for validation. For the DL, a pre-trained implemented the open-source DL framework was used to build and train the DL models transfer learning (Jia et al., 2014). AlexNet is a convolutional neural network constructed by the University of Toronto (Krizhevsky et al., 2012). The fundamental architecture of this CNN constituted eight pre-trained layers, including five convolutional/pooling that convolution of feature volume and reduces layers by compressing images using max pooling compresses by selecting the maximum value in each region as a representative value convolutional/pooling layer I converts the previous volume (224 × 224 × 3) to (11 × 11 × 3) convolutional/pooling layer II converts the result of layer I to (5 × 5 × 48) convolutional/pooling layer III converts the result of layer II to (3 × 3 × 256) convolutional/pooling layer IV converts the result of layer III to (3 × 3 × 192) convolutional/pooling layer V converts the result of layer IV to (3 × 3 × 192) fully-connected layers that make final connections between feature values and force to zero to suppress overfitting (dropout) total 4,096 neurons. Since AlexNet has 60 million parameters, their optimization was essential to avoid overfitting (**Figure 3**; Krizhevsky et al., 2012; Szegedy et al., 2014; Cagli et al., 2017; Rawat and Wang, 2017; Aggarwal et al., 2018; Ha et al., 2018; Vakli et al., 2018). The non-saturating nonlinearity f (x) = max (0, x) as the function instead of such as sigmoid function f (x) = $(1+e^{-x})^{-1}$ or f (x) = tanh (x) because the training time with gradient descent ReLUs much faster than that associated with if the input is negative, there is no contribution to other units (Nair and Hinton, 2010; Krizhevsky et al., 2012; Elfwing et al., 2018; Saha et al., 2018; Wang S. H. et al., 2018). Furthermore, adding a layer of local response normalization (LRN) between the pooling layer and the convolutional layer increases accuracy. The LRN is capable of handling a large number of CNNs with a large learning capacity that can be controlled by varying their assumptions about the nature of images that (1) the locality of pixel dependencies and (2) the stationarity of statistics.

The loss, which is a summation (not a percentage) of the errors in each dataset as shown below cross entropy error (CEE) with respect to the model's parameters by changing the weight vector values, in construction of the prediction models is calculated on training and validation datasets, where pi and yi correspond to the accuracy label (ground truth vector) and output of softmax (estimate values taken direct from the last layer output) for class *i*, respectively.

$$\mathrm{CEE} = -\Sigma\ (\mathrm{pi}) \log(\mathrm{yi})$$

The loss value implies how well or poorly a certain model behaves after each iteration of optimization. Loss is indicative of unless the model has over-fitted with respect to the training data. The accuracy of the model is usually determined after the validation samples are fed to the model and the number of mistakes (zero-one loss) that the model makes recorded. The percentage of misclassification is calculated (Martinez and Stiefelhagen, 2018; Nguyen et al., 2018; Zhang and Sabuncu, 2018; Khened et al., 2019).

Evaluation of the Predictive Models

In this method, it is possible to calculate the prediction result for each of a plurality of images prepared from the x-, y-, and z-axis directions with respect to one molecule. Therefore, the

[1] Jmol: An Open-Source Java Viewer for Chemical Structures in 3D. Available online at: http://www.jmol.org/

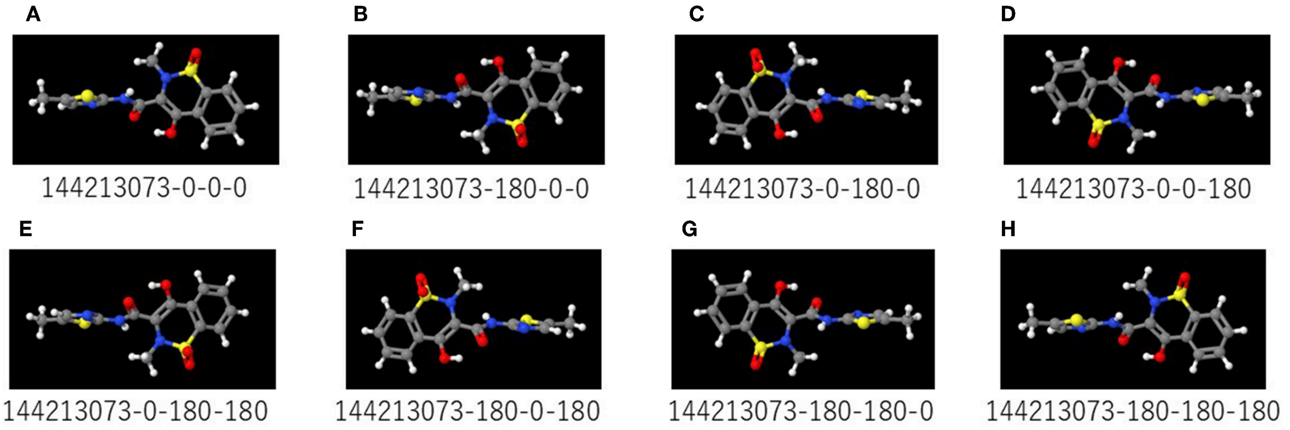


FIGURE 2 | (A–H) are representative images captured by rotating the 3D structure in 180° increments on Deep Snap. The numbers below the images are the substance identification numbers (SID) provided in the PubChem database and increments of the viewing direction on the x-, y-, and z-axes. Red, yellow, blue, white, and gray colors in the molecular structures indicate the oxygen, sulfur, nitrogen, hydrogen, and carbon atoms, respectively.

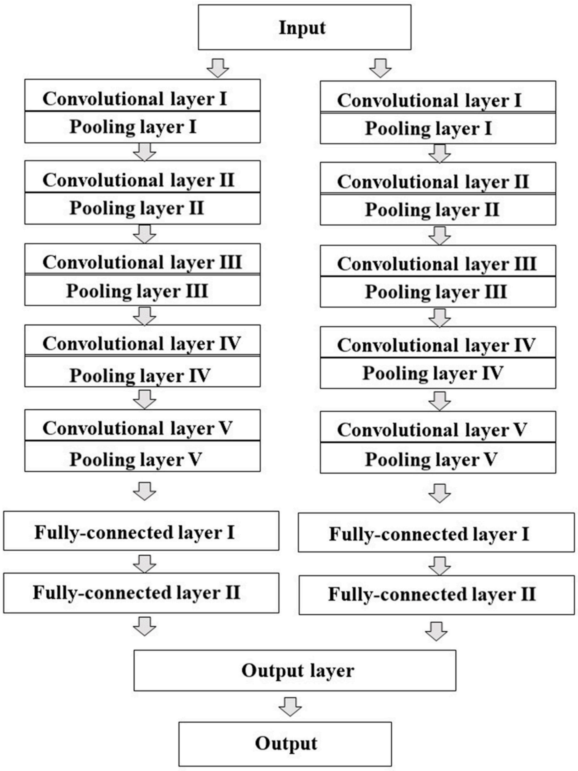


FIGURE 3 | Schematic representation of the architecture of the convolutional neural network (CNN) model. AlexNet was used as transfer learning. The CNN contains total eight pre-learned layers five convolutional and pooling layers automatically extracted features from input pixel data and three fully-connected layers. The two juxtaposed convolutional and pooling layers are finally combined to the third fully-connected layers.

median of all these predicted values generated per molecule was used as a representative predicted value for each molecule. The metric was calculated on the basis of the predicted and the experimentally determined (true) labels, and the auroc (area under receiver operating characteristic) was calculated using JMP pro 14, statistical discovery software (SAS Institute Inc. NC) to evaluate the predictive models using 3D chemical structures including training (38,088 pictures), validation (19,048 pictures), and testing (19,048 pictures) datasets captured from eight increments on the x-, y-, and z-axes: (0,0,0), (180,0,0), (0,180,0), (0,0,180), (0,180,180), (180,0,180), (180,180,0), and (180,180,180) (**Supplementary Table 2**) (Linden, 2006). Sensitivity describes the true positive rate i.e., the proportion of actual positive samples that were correctly identified as positive for all positive samples including true and false positives.

$$\text{Sensitivity} = \Sigma\ \text{True Positives}/(\Sigma\ \text{True Positives} + \Sigma\ \text{False Positives})$$

Specificity is the true negative rate i.e., the proportion of actual negative samples that were correctly identified as negative for all negative samples including true and false negatives.

$$\text{Specificity} = \Sigma\ \text{True Negatives}/(\Sigma\ \text{True Negatives} + \Sigma\ \text{False Positives})$$

Random Forest

The file, including chemical structures as indicated by SMILES, chemical annotation numbers, activity scores, dataset classes divided into training and validation. Based on this information, the 3D chemical structures were built, descriptors were calculated using the MOE chemical calculation system. Using these descriptors, the prediction model was constructed using JMP pro 14.

RESULTS AND DISCUSSION

The predictive models for the presence or absence of activity as a CAR agonist and cell viability were built using the open-source Caffe in combination with the Deep Snap approach were applied to the training (38,088 pictures) and validation (19,048 pictures) datasets 180° angle (**Supplementary Table 2**). The testing dataset (19,048 pictures) was used to measure the performance of each prediction model (**Supplementary Table 2**). The AUC was calculated. The correlations (R^2 values) of the AUC with each epoch were 0.95 (**Figure 4A**). The correlations (R^2 values) were calculated from the testing datasets with validation loss (VL), training loss (TL), and validation accuracy (VA). VL is an error summation not a percentage obtained from how well the model is doing for. TL is an error summation which by attempting to determine good values for all the weights and biases (an empirical risk minimization). VA is the percentage of correct answers based on the results obtained from. As results, these R^2 values with AUCs were 0.86 (VL), 0.62 (TL), and 0.57 (VA), respectively (**Figures 4B–D**). Moreover, the R^2 values of the VL, TL, and VA each epochs were 0.90, 0.65, and 0.61, respectively (**Figures 4E–G**). These findings suggest that VL is the most important parameter of those considered here for evaluating the performance of a DL model.

Next, the parameters for capturing Jmol-generated images on Deep Snap were optimized by assessing the DL models using the same procedure based on the VL using four pictures on the x-, y-, and z-axes: (0,0,0), (180,0,0), (0,180,0), and (0,0,180) in the training (19,044 pictures), validation (9,524 pictures), and test (9,524 pictures) datasets (**Figures 2A–D** and **Supplementary Table 2**). The following parameters were considered: (1) the number of molecules per SDF file: MPS, (2) the zoom factor: ZF, (3) the atom size for Van der Waals interactions: AT, (4) the bond radius: BR, (5) the minimum bond distance: MBD, and (6) the bond tolerance: BT. The parameter values (and corresponding minimum VL values) for the best model are as follows: (1) MPS: 150 (0.430), (2) ZF: 80% (0.431), (3) AT: 22% (0.435), (4) BR: 20 mÃ (0.425), (5) MBD: 0.4 Ã (0.430), and (6) BT: 0.8 Ã (0.436) (**Figures 5A–F**). In addition, the R^2 values between these parameters and VLs were more than 0.90, and each of these relations followed quadratic function curves. Also, the R^2 values of the running time (RT) in DL with the above six parameters showed that the RTs were moderately associated with AT ($R^2 = 0.48$), BR ($R^2 = 0.47$), and BT ($R^2 = 0.43$) (**Supplementary Figures 1C,D,F**). However, MPS, ZF, and MBD showed no associations (**Supplementary Figures 1A,B,E**). Similarly, the image pixel size (IPS) was examined in the same way as the VL and RT in DL using three pictures on the x-, y-, and z-axes: (0,0,0), (0,90,0), and (0,0,90) in the training (14,283 pictures, 4,761 compounds), validation (7,143 pictures, 2,381 compounds), and test (7,143 pictures, 2,381 compounds) datasets (**Supplementary Table 2**). The IPSs (256×256) and (64×64) exhibited minimum VL (0.440) (**Figure 6A**) and minimum RT (10 min) (**Figure 6B**), respectively. Moreover, the number of cores in the multi-core CPU architecture showed the minimum RT (8 min) in the Jmol-generated images with 70 (**Figure 6C**). Also, we explored the effects of the minimum VL with space-filling, where the atoms are represented by spheres whose radii and center-to-center distances are proportional to the radii of the atoms and the distances between the atomic nuclei using one (0,0,0) or four (0,0,0), (180,0,0), (0,180,0), (0,0,180) image angles (**Figures 2A–D**) on the optimized parameters. When using one image, space-filling chemical structures into the image slightly increased the minimum VL (0.456) compared with that of normal spacing (0.452) (**Figure 6D**, left). However, there were no minimum VL changes between space-filling and normal spacing when using four image angles (**Figure 6D**, right). Furthermore, we compared the influence of the image color types of chemical structures with the minimum VL by using one or four image angles the optimized parameters, similarly. When the atomic colors of all the structures were changed to monotone (gray or white), these minimum VLs (0.468 or 0.467 for gray and white, respectively) increased to more than that of normal multi-color structures (0.442) using four image angles (**Figure 6E**, right). However, in the structures where the color of all atoms was changed to gray except for hydrogen (two-color: gray + white), the minimum VL (0.437) was decreased slightly compared with that of normal multi-color structures (0.442) using the four images (**Figure 6E**, right). When one angle image was used similarly, increased minimum VL of gray (0.499), white (0.468), or gray + white (0.460) was observed compared with that of normal multi-color (0.455) (**Figure 6E**, left). These findings suggest that optimal thresholds exist to attain the best performance with the prediction model. Finally, using the parameter values for the best performance model, AUCs were calculated using eight images of chemical structures captured at 180° increments on the x-, y-, and z-axes. As a result of optimization, the AUC exhibited 0.764 with minimum VL of 0.432. Furthermore, using 64 images at 105° angle and with default parameter values other than BR 15mÃ, the AUC increased into 0.791.

To assess (1) the suitableness of input as supervised data, (2) sufficient amount of images for training, and (3) adequate training for input dataset of pictures of chemical structure into the DL, the activity scores of the datasets, including training, validation, and test, were randomly assigned keeping the numbers of the three datasets unchanged as permutation test. The calculation of the performed each parameterized values of Deep Snap with each best performance model to capture chemical structures eight pictures at 180° angle using a total of ten different datasets with assignments of various activity scores. As result, the average AUCs were 0.553 (±0.007) with the average minimum VL of 0.522 (±0.014), indicated almost random guessing. These results suggest that the prediction models in this study extracted the CAR agonist activity-specific structural features from chemical compounds. Also, we calculated the AUC random forest as another method the same datasets for the above Deep Snap for CAR agonist and 206 of descriptors to build the prediction model in ROC-AUC value 0.749. Previously, we found that the prediction for the performance of compounds inducing MMP disruption was better 45° angles using 512 pictures for one molecule, with AUCs of 0.921 (Uesawa, 2018). Moreover, using 90° angle which 64 pictures for each, the performance of the prediction model indicated that the ROC-AUC value was 0.898 (Uesawa, 2018). In this study, we have used only 64 pictures based on 105° angle to avoid high computational cost. These results suggested that the prediction performance in the Deep Snap-DL method could be improved by input images due to more information about chemical structures. Also, as for the score

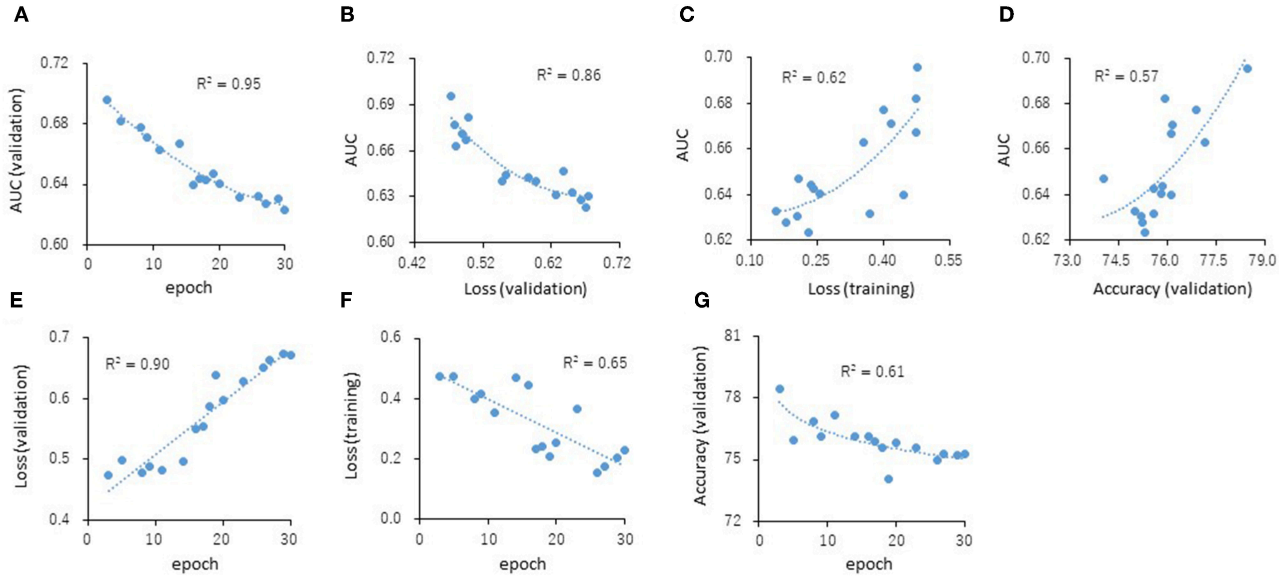


FIGURE 4 | Correlations of the epochs **(A)**, validation loss **(B)**, training loss **(C)**, and validation accuracy **(D)** with the AUCs and the validation loss **(E)**, training loss **(F)**, and validation accuracy **(G)** with the epochs. The R^2 values represent the correlation coefficients with two-dimensional equation representing the approximate fitted curve.

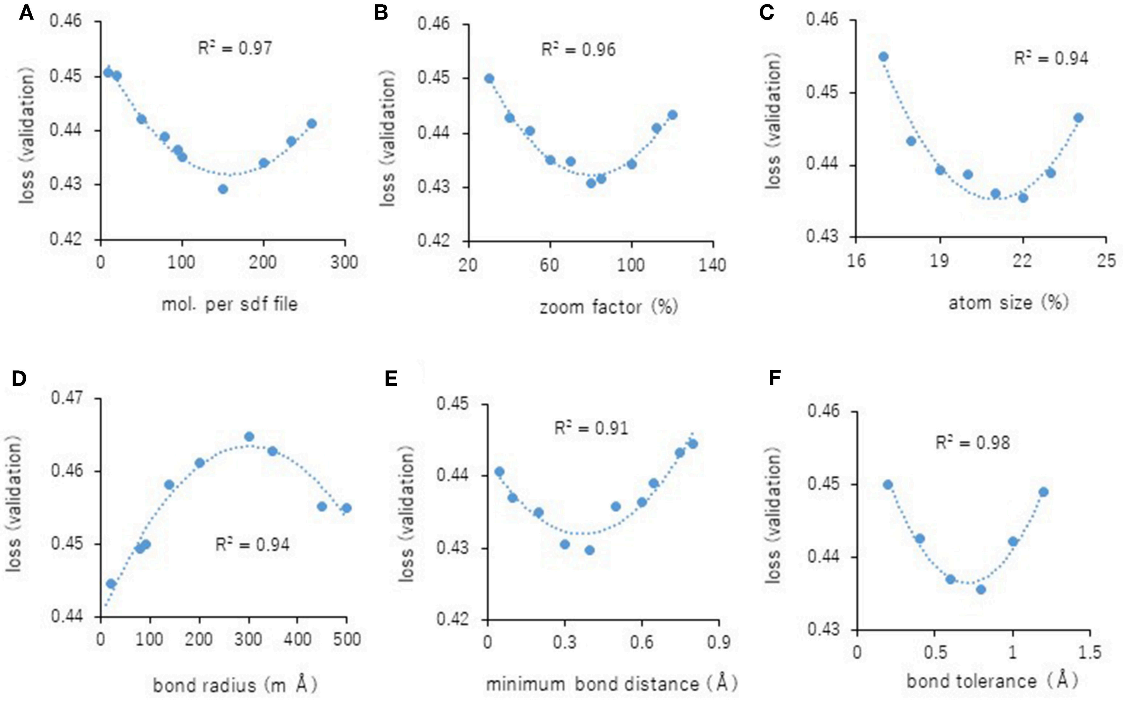


FIGURE 5 | (A–F) displays parameterization of performance on Deep Snap. Correlation between the minimum VL of each epoch and the parameter values **(A)**: MPS, **(B)**: ZF, **(C)**: AS, **(D)**: BR, **(E)**: MBD, and **(F)**: BT for four images based on the 180° angle.

activity of the CAR, the chemicals with scores other than 0 were defined as positive in order to secure enough input data in this study. However, in Tox21 program, the obvious activity for the CAR agonist is defined for chemicals with score of more than 40 (PubChem; https://pubchem.ncbi.nlm.nih.gov/#, AID 1224892). Therefore, it is necessary to optimize various types of assignments for the activity scores and/or other datasets in detail to further increase the prediction performance. In addition, a comparison

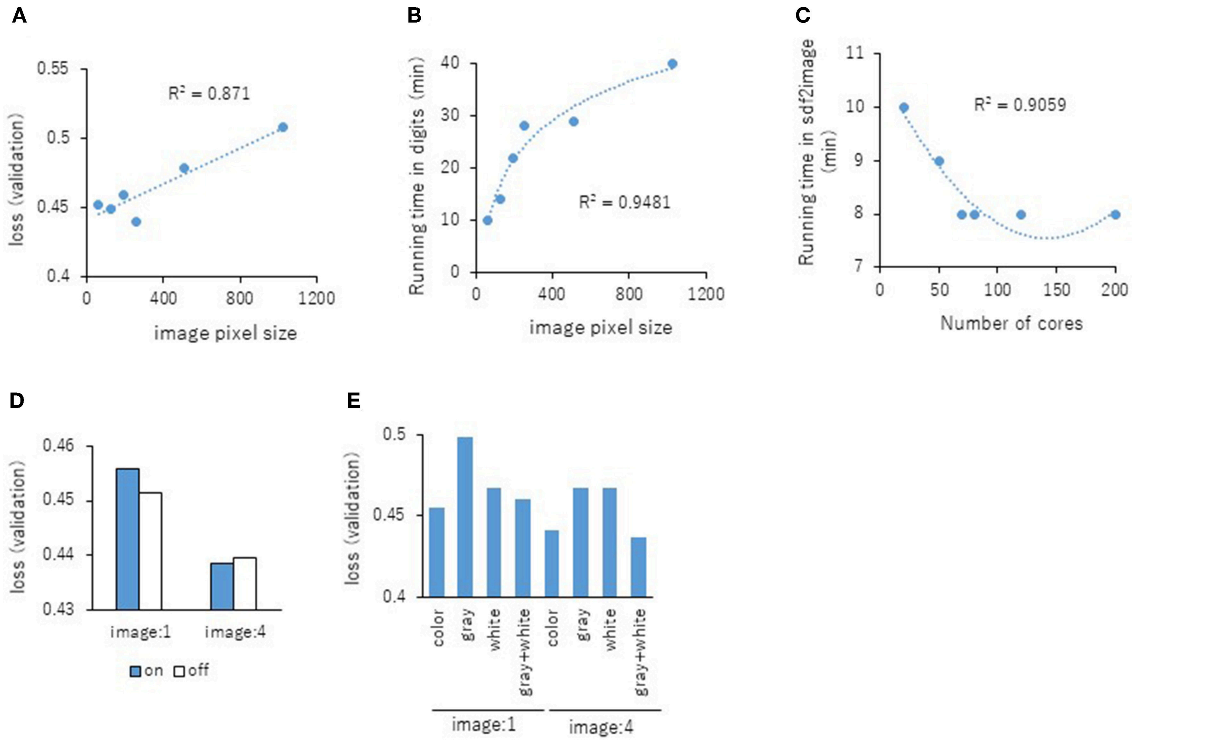


FIGURE 6 | Relationship between the IPS and the minimum VL of each epochs **(A)** or RT in DL **(B)** using three pictures on the angle of 90° with R^2 values between the IPS and the minimum VL or RT. **(C)** Influence of RT in three images with the number **(D)** The minimum VLs of space-filling (on; blue bar) and normal spacing (off; white bar) using one or four angles images. **(E)** The minimum VLs of multi-color, monotone-color (gray and white), and two-color (gray + white) using one or four angles images.

of the performances between this state-of-the-art Deep Snap and 1,024 of extended-connectivity fingerprint (ECFG) of descriptors calculated from Dragon 7.0 (Kode srl., Pisa, Italy, Rogers and Hahn, 2010; Nikolic et al., 2012; Concu and Cordeiro, 2018; Uesawa, 2018). The prediction model constructed by DL in an H2O 3.2 package, where hidden layers, epochs, and best epochs were 200, 10, and 5, respectively (H20 ai, CA, USA, Chow, 2014) with ECFP showed that the ROC-AUC was 0.888 (Uesawa, 2018). In addition, the random forest in JMP pro 14, in which number of terms and maximum splits per tree were 500 and 256 for fingerprint, and 500 and 29 for 3D descriptors, respectively, predicted the models using the above ECFP descriptors or 3D descriptors with AUC of 0.901 or 0.907 (Uesawa, 2018). Until today, to improve the performance of prediction model, the selection of structural descriptors carried out using the skills and knowledge. Because it is difficult to perfectly preserve the original data, many of these descriptors are irreversible conversions. However, in the DL method using task-specific automatically extracted image information for molecular structures that do not require such high craftsmanship input data, it may demonstrate equal to or better than the above method using descriptors hand-engineered without prior knowledge or assumptions about the features.

When considering applying DL to a compound, whose molecular structure is a variable data format that can have branches and loops, there are problems with how to handle that input or output. To address this issue, graphic-based convolution, which has the ability to handle graph structures, simple encoding of the molecules (atoms, bonds, distances, etc.) represented by edge-connected nodes introducing convolution operations on each nodes non-Euclidean structure was proposed as modifications of DL architectures specialized for molecular fingerprints and models in the terms of structural features, physical properties, and activity (Duvenaud et al., 2015; Gilmer et al., 2017; Zhou and Li, 2017; Fernandez et al., 2018; Li C. et al., 2018). Since a chemical compound can also be represented as an undirected graphs of atoms when an atom is defined as a vertex (node) and a bond is defined as a side (edge), it is possible to construct a highly accurate prediction model by applying a convolution operation to the graph including their physical and chemical properties and extracting meaningful features from the large scale datasets of graph structure (Defferrard et al., 2016; Kipf and Welling, 2016). However, unlike image data, there drawback that a connection relation of peripheral nodes around the attention node of the graph is indefinite for each target node. To solve this difficulty with a heuristic or theoretical approach, graph convolution can be applied to graph Fourier transformation considering the adjacency of nodes by parameterizing weighted and undirected graphs without loops and multiple edges. Fourier conversion decomposes a waveform signal component by frequency component, but graph Fourier conversion decomposed a signal defined on a graph into "gentle signal" or "steep signal." As for chemical structure, the graph signal converts into a graph spectral region assigning feature vectors to each atom in a chemical substance and their interaction between atoms. Thus, it is very well-adapted to prediction of local molecular structure-dependent physiological activity. In the case of definitions derived from the graph Fourier transform, for technical reasons, it needs to undirected and weighted graph without loops and multiple edges. On the other hand, by defining graph convolution more directly from only the connection relationship of nodes and edges, it is possible to introduce a more complicated structure such as a directed graph, multiple edges, and loops to graph convolution (Schlichtkrull et al., 2017). That is, for each node, its adjacent nodes are classified according to how they are connected, and then the sum (or average) of the signals of the neighboring nodes is taken for each neighborhood according to the manner of connection and according to how it is connected. However, since this method relied on edge and/or node information, the graph structures from the 3D conformational flexibility and the diversity of many features on the edge and/or node, such as shape, electrostatics, quantum effects, and other properties emerged from the molecular graph essential to clearly represent the biological systems and their relationship for the molecular activity and to consistently outperform other models (Kearnes et al., 2016). Additionally, since this graph structured format is heterogeneous among molecules, many learning algorithms how to process the complex graph effectively, except homogeneous input features. Therefore, to resolve issues, data transformings for the graph structure of the molecules to fix data size and format (Duvenaud et al., 2015; Liu K. et al., 2018). In addition, representations by the SMILES (Weininger, 1988; Putz and Dudaş, 2013; Achary, 2014; Jastrzebski et al., 2018; Kumar and Chauhan, 2018) do not encode bond lengths and mutual orientation of atom in space, meaning that they lack information for the molecular conformations, such as 3D atomic arrangements and some molecule stereoisomers.

Also, 3D-CNN, convolutional layers extended to 3D filter that move 3-directions (x, y, z) extract spatiotemporal features from moving objects proposed as a method applied to motion image recognition (Ji et al., 2013; Blendowski and Heinrich, 2018; Lu et al., 2018). It has been successfully used to extract against the temporal change of the spatial structure data as a feature expression of 3D volume space such as cuboid output using the node locally connected to all the images within a certain time width (Ji et al., 2013; Maturana and Scherer, 2015). In this method, although the temporal change such as event detection in videos, 3D images etc. is considered in the extracted feature, it depends on the size in the time direction of the filter. Therefore, when recognizing an operation longer than the filter size, selection and combination processing of those features must be performed. As for chemical compounds, the 3D-CNN has been successfully shown to able to handle the data with spatial structure such as 3D-structures, on the choice of the data representation (Ji et al., 2013; Maturana and Scherer, 2015; Blendowski and Heinrich, 2018; Kuzminykh et al., 2018). If a suitable representation used, the most critical information efficiently captured. In addition, the chemical compounds induced conformational changes target interactions is possible to a number of conformations or orientations (Tuffery and Derreumaux, 2017; Salmaso and Moro, 2018). Furthermore, the conformational changes of target proteins by ligands and protein-ligands interactions have been studied computational (Yang et al., 2016; Hollingsworth and Dror, 2018; Nusrat and Khan, 2018). Therefore, the 3D-CNN could be a very useful method for extracting structural features based on molecular dynamics, which the dynamic behavior of molecular system as a function of time. However, since a data in non-euclidean spaces, such as spherical data is difficult to trivially apply for direct 3D representation, the suitable conditions such as scaling and required number of input samples have not been cleared completely, which leads to poor performance by sparsity and redundancy in the data and increased complexity in the convolution process (Ji et al., 2013; Maturana and Scherer, 2015; Blendowski and Heinrich, 2018; Kuzminykh et al., 2018). In additions, 3D-CNNs requires more 3D matrix and more calculations than 2D. Thus, the scaling for the CNNs to 3D representations is not straightforward due to the sparsity in input data and the complexity in the convolution operations (Ji et al., 2013; Maturana and Scherer, 2015; Blendowski and Heinrich, 2018; Kuzminykh et al., 2018). Therefore, even now, 3D-CNN need shape descriptors by hand, such as light field descriptors (Pu and Ramani, 2006), mesh DOG (Zaharescu et al., 2009), spin images (Johnson and Hebert, 1999), heat kernel signatures (Xiang et al., 2014), and spherical harmonics high performance (Kazhdan et al., 2003). To alleviate this problem, although Gaussian blur representation was proposed to reduce the sparsity and the redundancy of input, convolving with the Gaussian kernel leads to information loss (Kuzminykh et al., 2018).

Previously, it was ascertained that the Deep Snap-DL method yields the corresponding predicted values for different physiological activities between optical R/S isomers (Uesawa, 2018). This report indicated that Deep Snap-DL accurately extract physiological activities depending on molecular conformation-specificity optimization for various conformations is necessary to maintain high performance of the prediction model. In this research, to define the steric conformation of the molecular structure, CORINA Classic software was used. However, if more suitable definition of 3D steric structures of chemical compounds directly or indirectly related to biological activity, mechanisms, and molecular pathways such as determination of 3D structure for a protein receptor with apparent ligand affinity pocket were established based on the molecular dynamics stimulation, the Deep Snap-DL procedure would be outperformed.

On the other hand, there are some problems that need to be improved so far in this Deep Snap-DL method. At first, in principle, this strategy to capture more detail and greater amount of information chemical structures using more molecular images from 3D-rotation (Uesawa, 2018). In supervised learning, output data corresponding to input data can be obtained, but learning is performed for the purpose of minimizing the error by comparing the output to new data. Therefore, the correction of misclassification for a large amount of labeled input data is difficult. If the classification criteria within image data could be clarified using proposed visual explanations technique (Simonyan et al., 2013; Mahendran and Vedaldi, 2014; Selvaraju et al., 2016; Smilkov et al., 2017; Zhen et al., 2017; Philbrick et al., 2018), it may be useful for estimation of 3D structure important for physiological activity of the compound and would more reduction of calculation cost by reducing the number of images used. Furthermore, by parameters for Deep Snap in this study, the calculation time was reduced the relatively high performance of the prediction model for the CAR agonist activity. In agreement with previous report although DL able to accurately predict for a molecule with just close neighbors in the training dataset, a hitherto unexamined chemical was predicted close to the average of all training molecule activities, which the lack of ability to learn beyond the training dataset (Liu R. et al., 2018). Deep Snap-DL method indicated the performances of prediction models depending on input datasets produced by various conditions including bonds, spacing, angles, colors, atom size, etc. Moreover, the AUCs were reduced by random permutation of the activity scores of datasets consisting training, validations, and test as non-endpoint activity. These findings suggested that the task-specific improvement of Deep Snap-DL technique by adjustments of input data with the representations of chemical structure such as bonds, space, atom size etc. could be more available approach than conventional methods. Taken together, by combining the Deep Snap strategy with parts of graph-CNN or 3D-CNN functions. Overall, the novel approach Deep Snap not only would fill a gap between chemical structure and toxicological prediction, but also may be useful for constructing an *in silico* prediction model of appropriate chemical risk assessment replace.

In summary, the relations of the parameters of Deep Snap such as (1) number of molecules per SDF files split into (2) zoom factor percentage, (3) atom size for van der waals percentage, (4) bond radius, (5) minimum bond distance, and (6) bond tolerance with the VLs as indicator for evaluating the performance of the DL following quadratic function curves, suggesting that optimal thresholds exist to attain the best performance with these prediction models. Using the parameter values the best performance with the prediction model, the prediction model for CAR agonist was built using 64 images at 105° angle AUCs of 0.791. The results of this study feature the possible power of novel DL-based QSAR approach for prediction of potential toxicity of large datasets of any chemical compounds.

AUTHOR CONTRIBUTIONS

YU initiated and supervised the work, designed the experiments, collected the information about chemical compounds, and edited the manuscript. YM drafted the manuscript. YU and YM read and approved the final manuscript.

FUNDING

This study is supported in part by grants from Long-Range Research Initiative, Japan Chemical Industry Association (16_PT01-02) and Ministry of Economy, Trade and Industry, AI-SHIPS (AI-based Substances Hazardous Integrated Prediction System) project (20180314ZaiSei8).

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fbioe.2019.00065/full#supplementary-material

REFERENCES

3D Structure Generator CORINA Classic. (2019). *3D Structure Generator CORINA Classic*. Nürnberg: Molecular Networks GmbH. Available online at: www.mn-am.com

Abdelaziz, A., Spahn-Langguth, H., Schramm, K -W., and Tetko, I. V. (2016). Consensus modeling for HTS assays using *in silico* descriptors calculates the best balanced accuracy in Tox21 challenge. *Front. Environ. Sci.* 4:2. doi: 10.3389/fenvs.2016.00002

Achary, P. G. (2014). Simplified molecular input line entry system-based optimal descriptors: QSARmodelling for voltage-gated potassium channel subunit Kv7.2. *SAR QSAR Environ. Res.* 25, 73–90. doi: 10.1080/1062936X.2013.842930

Aggarwal, H. K., Mani, M. P., and Jacob, M. (2018). MoDL: model based deep learning architecture for inverse problems. *IEEE Trans. Med. Imaging.* 38, 394–405. doi: 10.1109/TMI.2018.2865356

Ambe, K., Ishihara, K., Ochibe, T., Ohya, K., Tamura, S., Inoue, K., et al. (2018). *In silico* prediction of chemical-induced hepatocellular hypertrophy using molecular descriptors. *Toxicol. Sci.* 162, 667–675. doi: 10.1093/toxsci/kfx287

Andronico, A., Randall, A., Benz, R. W., and Baldi, P. (2011). Data-driven high-throughput prediction of the 3-D structure of small molecules: review and progress. *J. Chem. Inf. Model.* 51, 760–776. doi: 10.1021/ci100223t

Attene-Ramos, M. S., Huang, R., Michael, S., Witt, K. L., Richard, A., Tice, R. R., et al. (2015). Profiling of the Tox21 chemical collection for mitochondrial function to identify compounds that acutely decrease mitochondrial membrane potential. *Environ. Health Perspect.* 123, 49–56. doi: 10.1289/ehp.1408642

Attene-Ramos, M. S., Miller, N., Huang, R., Michael, S., Itkin, M., Kavlock, R. J., et al. (2013). The Tox21 robotic platform for the assessment of environmental chemicals from vision to reality. *Drug Discov. Today.* 18, 716–723. doi: 10.1016/j.drudis.2013.05.015

Azimi, S. M., Britz, D., Engstler, M., Fritz, M., and Mücklich, F. (2018). Advanced steel microstructural classification by methods. *Sci. Rep.* 8:2128. doi: 10.1038/s41598-018-20037-5

Banerjee, P., Eckert, A. O., Schrey, A. K., and Preissner, R. (2018). ProTox-II: a webserver for the prediction of toxicity of chemicals. *Nucleic Acids Res.* 46, W257–W263. doi: 10.1093/nar/gky318

Bell, S. M., Phillips, J., Sedykh, A., Tandon, A., Sprankle, C., Morefield, S. Q., et al. (2017). An integrated chemical environment to support 21st-century toxicology. *Environ. Health Perspect.* 125:054501. doi: 10.1289/EHP1759

Bengio, Y., Courville, A., and Vincent, P. (2013). Representation learning: a review and new perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.* 35, 1798–1828. doi: 10.1109/TPAMI.2013.50

Blendowski, M., and Heinrich, M. P. (2018). Combining MRF-based deformable registration and deep binary 3D-CNN descriptors for large lung motion estimation in COPD patients. *Int. J. Comput. Assist. Radiol Surg.* 14, 43–52. doi: 10.1007/s11548-018-1888-2

Bloomingdale, P., Housand, C., Apgar, J. F., Millard, B. L., Mager, D. E., Burke, J. M., et al. (2017). Quantitative systems toxicology. *Curr. Opin. Toxicol.* 4, 79–87. doi: 10.1016/j.cotox.2017.07.003

Bolisetty, S., Traylor, A., Zarjou, A., Johnson, M. S., Benavides, G. A., Ricart, K., et al. (2013). Mitochondria-targeted heme oxygenase-1 decreases oxidative stress in renal epithelial cells. *Am. J. Physiol. Renal. Physiol.* 305, F255–F264. doi: 10.1152/ajprenal.00160.2013

Brockmeier, E. K., Hodges, G., Hutchinson, T. H., Butler, E., Hecker, M., Tollefsen, K. E., et al. (2017). The role of omics in the application of adverse outcome pathways for chemical risk assessment. *Toxicol. Sci.* 158, 252–262. doi: 10.1093/toxsci/kfx097

Cagli, E., Dumas, C., and Prouff, E. (2017). *Convolutional Neural Networks with Data Augmentation against Jitter-Based Countermeasures—Profiling Attacks without Pre-Processing*. Cryptology ePrint Archive: Report 2017/740.

Cammer, S. (2007). SChiSM2: creating interactive web page annotations of molecular structure models using Jmol. *Bioinformatics.* 23, 383–384. doi: 10.1093/bioinformatics/btl603

Chen, I. J., and Foloppe, N. (2008). Conformational sampling of druglike molecules with MOE and catalyst: implications for pharmacophore modeling and virtual screening. *J. Chem. Inf. Model*. 48, 1773–1791. doi: 10.1021/ci800130k

Chen, S., Hsieh, J. H., Huang, R., Sakamuru, S., Hsin, L. Y., Xia, M., et al. (2015). Cell-based high-throughput screening for aromatase inhibitors in the Tox21 10K library. *Toxicol. Sci.* 147, 446–457. doi: 10.1093/toxsci/kfv141

Chow, J -F. (2014). *Things to Try After useR!—Part 1: Deep Learning with H2O*. Available online at: http://www.r-bloggers.com/things-to-try-after-user-part-1-deeplearning-with-h2o/ (Accessed August 10, 2017).

Cipullo, S., Snapir, B., Prpich, G., Campo, P., and Coulon, F. (2019). Prediction of bioavailability and toxicity of complex chemical mixtures through machine learning models. *Chemosphere* 215, 388–395. doi: 10.1016/j.chemosphere.2018.10.056

Clark, M., and Steger-Hartmann, T. (2018). A big data approach to the concordance of the toxicity of pharmaceuticals in animals and humans. *Regul. Toxicol. Pharmacol.* 96, 94–105. doi: 10.1016/j.yrtph.2018.04.018

Collins, F. S., Gray, G. M., and Bucher, J. R. (2008). Toxicology. Transforming environmental health protection. *Science* 319, 906–907. doi: 10.1126/science.1154619

Concu, R., and Cordeiro, M. N. D. S. (2018). Looking for new inhibitors for the epidermal growth factor receptor. *Curr. Top. Med. Chem.* 18, 219–232. doi: 10.2174/1568026618666180329123023

Defferrard, M., Bresson, X., and Vandergheynst, P. (2016). Convolutional neural networks on graphs with fast localized spectral filtering. *arXiv:1606.09375v3* [Preprint]. Available online at: https://arxiv.org/pdf/1606.09375.pdf

Dougall, L. G. (2001). Functional methods for quantifying agonists and antagonists. *J. Recept. Signal Transduct. Res.* 21, 117–137. doi: 10.1081/RRS-100107425

Duvenaud, D., Maclaurin, D., Aguilera-Iparraguirre, J., Gómez-Bombarelli, R., Hirzel, T., Aspuru-Guzik, A., et al. (2015). Convolutional networks on graphs for learning molecular fingerprints. *arXiv:1509.09292v2*. Available online at: https://arxiv.org/pdf/1509.09292.pdf

Elfwing, S., Uchibe, E., and Doya, K. (2018). Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural. Netw.* 107, 3–11. doi: 10.1016/j.neunet.2017.12.012

Fang, H., Tong, W., Welsh, W. J., and Sheehan, D. M. (2003). QSAR models in receptor mediated effects: the nuclear receptor superfamily. *J. Mol. Struct.* 622, 113–125. doi: 10.1016/S0166-1280(02)00623-1

Fang, X., Bagui, S., and Bagui, S. (2017). Improving virtual screening predictive accuracy of Human kallikrein 5 inhibitors using machine learning models. *Comput. Biol. Chem.* 69, 110–119. doi: 10.1016/j.compbiolchem.2017.05.007

Fay, K. A., Villeneuve, D. L., Swintek, J., Edwards, S. W., Nelms, M. D., Blackwell, B. R., et al. (2018). Differentiating pathway-specific from nonspecific effects in high-throughput toxicity data: a foundation for prioritizing adverse outcome pathway development. *Toxicol. Sci.* 163, 500–515. doi: 10.1093/toxsci/kfy049

Fernandez, M., Ban, F., Woo, G., Hsing, M., Yamazaki, T., LeBlanc, E., et al. (2018). Toxic colors: the use of deep learning for predicting toxicity of compounds merely from their graphic images. *J. Chem. Inf. Model.* 58, 1533–1543. doi: 10.1021/acs.jcim.8b00338

Gawehn, E., Hiss, J. A., Brown, J. B., and Schneider, G. (2018). Advancing drug discovery via GPU-based deep learning. *Expert Opin. Drug Discov.* 13, 579–582. doi: 10.1080/17460441.2018.1465407

Genuis, S. J., and Kyrillos, E. (2017). The chemical disruption of human metabolism. *Toxicol. Mech. Methods.* 27, 477–500. doi: 10.1080/15376516.2017.1323986

Ghasemi, F., Mehridehnavi, A., Pérez-Garrido, A., and Pérez-Sánchez, H. (2018). Neural network and deep-learning algorithms used in QSAR studies: merits and drawbacks. *Drug Discov. Today.* 23, 1784–1790. doi: 10.1016/j.drudis.2018.06.016

Gilmer, J., Schoenholz, S. S., Riley. P. F., VInyals, O., and Dahl, G. E. (2017). Neural message passing for quantum chemistry. *arXiv:1704.01212v2* [Preprint]. Available online at: https://arxiv.org/pdf/1704.01212.pdf

Guimarães, M. C., Duarte, M. H., Silla, J. M., and Freitas, M. P. (2016). Is conformation a fundamental descriptor in QSAR? A case for halogenated anesthetics. *Beilstein J. Org. Chem.* 12, 760–768. doi: 10.3762/bjoc.12.76

Ha, R., Chang, P., Karcich, J., Mutasa, S., Fardanesh, R., Wynn, R. T., et al. (2018). Axillary lymph node evaluation utilizing convolutional neural networks using MRI dataset. *J. Digit Imaging.* 31, 851–856. doi: 10.1007/s10278-018-0086-7

Halder, A. K., Moura, A. S., and Cordeiro, M. N. D. S. (2018). QSAR modelling: a therapeutic patent review 2010-present. *Expert Opin. Ther. Pat.* 28, 467–476. doi: 10.1080/13543776.2018.1475560

Hanson, R. M. (2016). Jmol SMILES and Jmol SMARTS: specifications and applications. *J. Cheminform.* 26:50. doi: 10.1186/s13321-016-0160-4

Hanson, R. M., and Lu, X. J. (2017). DSSR-enhanced visualization of nucleic acid structures in Jmol. *Nucleic Acids Res.* 45:W528–W533. doi: 10.1093/nar/gkx365

Heindel, J. J. (2018). The developmental basis of disease: Update on environmental exposures and animal models. *Basic Clin. Pharmacol. Toxicol.* 1–9. doi: 10.1111/bcpt.13118

Heindel, J. J., Skalla, L. A., Joubert, B. R., Dilworth, C. H., and Gray, K. A. (2017). Review of developmental origins of health and disease publications in environmental epidemiology. *Reprod. Toxicol.* 68, 34–48. doi: 10.1016/j.reprotox.2016.11.011

Herráez, A. (2006). Biomolecules in the computer: Jmol to the rescue. *Biochem. Mol. Biol. Educ.* 34, 255–261. doi: 10.1002/bmb.2006.494034042644

Hollingsworth, S. A., and Dror, R. O. (2018). Molecular dynamics simulation for all. *Neuron.* 99, 1129–1143. doi: 10.1016/j.neuron.2018.08.011

Hsieh, J. H., Huang, R., Lin, J. A., Sedykh, A., Zhao, J., Tice, R. R., et al. (2017). Real-time cell toxicity profiling of Tox21 10K compounds reveals cytotoxicity dependent toxicity pathway linkage. *PLoS ONE* 12:e0177902. doi: 10.1371/journal.pone.0177902

Hsieh, J. H., Sedykh, A., Huang, R., Xia, M., and Tice, R. R. (2015). A data analysis pipeline accounting for artifacts in Tox21 quantitative high-throughput screening assays. *J. Biomol. Screen.* 20, 887–897. doi: 10.1177/1087057115581317

Hu, G., Wang, K., Peng, Y., Qiu, M., Shi, J., and Liu, L. (2018). Deep learning methods for underwater target feature extraction and recognition. *Comput. Intell. Neurosci.* 2018:10. doi: 10.1155/2018/1214301

Hua, S., Zhang, H., Song, Y., Li, R., Liu, J., Wang, Y., et al. (2012). High expression of Mfn1 promotes early development of bovine SCNT embryos: improvement of mitochondrial membrane potential and oxidative metabolism. *Mitochondrion.* 12, 320–327. doi: 10.1016/j.mito.2011.12.002

Huang, R., Sakamuru, S., Martin, M. T., Reif, D. M., Judson, R. S., Houck, K. A., et al. (2014). Profiling of the Tox21 10K compound library for agonists and antagonists of the estrogen receptor alpha signaling pathway. *Sci Rep*. 4:5664. doi: 10.1038/srep05664

Huang, R., Southall, N., Wang, Y., Yasgar, A., Shinn, P., Jadhav, A., et al. (2011). The NCGC pharmaceutical collection: a comprehensive resource of clinically approved drugs enabling repurposing and chemical genomics. *Sci. Transl. Med.* 3:80ps16. doi: 10.1126/scitranslmed.3001862

Huang, R., and Xia, M. (2017). Editorial: Tox21 challenge to build predictive models of nuclear receptor and stress response pathways as mediated by exposure to environmental toxicants and drugs. *Front. Environ. Sci.* 5, 1–3. doi: 10.3389/fenvs.2017.00003

Huang, R., Xia, M., Sakamuru, S., Zhao, J., Shahane, S. A., Attene-Ramos, M., et al. (2016). Modelling the Tox21 10 K chemical profiles for *in vivo* toxicity prediction and mechanism characterization. *Nat Commun.* 7:10425. doi: 10.1038/ncomms10425

Hussain, Z., Gimenez, F., Yi, D., and Rubin, D. (2018). Differential data augmentation techniques for medical imaging classification tasks. *AMIA Annu. Symp. Proc.* 2017, 979–984.

Insel, P. A., Amara, S. G., Blaschke, T. F., and Meyer, U. A. (2017). Introduction to the theme "new methods and novel therapeutic approaches in pharmacology and toxicology". *Annu. Rev. Pharmacol. Toxicol.* 57, 13–17. doi: 10.1146/annurev-pharmtox-091616-023708

Jastrzebski, S., Leśniak, D., and Czarnecki, V. M. (2018). Learning to SMILE(S). *arXiv:1602.06289* [Preprint]. Available online at: https://arxiv.org/pdf/1602.06289.pdf

Ji, S., Yang, M., and Yu, K. (2013). 3D convolutional neural networks for human action recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* 35, 221–231. doi: 10.1109/TPAMI.2012.59

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., et al. (2014). Caffe: convolutional architecture for fast feature embedding. *CVPR* 675–678. Available online at: https://ucb-icsi-vision-group.github.io/caffe-paper/caffe.pdf

Johnson, A. E., and Hebert, M. (1999). Using spin images for efficient object recognition in cluttered 3D scenes. *IEEE Trans. Pattern Anal. Machine Intelligence* 21, 433–449. doi: 10.1109/34.765655

Johnson, R., and Zhang, T. (2015). Semi-supervised convolutional neural networks for text categorization via region embedding. *Adv. Neural. Inf. Process. Syst.* 28, 919–927. Available online at: https://papers.nips.cc/paper/5849-semi-supervised-convolutional-neural-networks-for-text-categorization-via-region-embedding.pdf

Juberg, D. R., Knudsen, T. B., Sander, M., Beck, N. B., Faustman, E. M., Mendrick, D. L., et al. (2017). FutureTox III: bridges for translation. *Toxicol. Sci.* 155, 22–31. doi: 10.1093/toxsci/kfw194

Kavlock, R. J., Austin, C. P., and Tice, R. R. (2009). Toxicity testing in the 21st century: implications for human health risk assessment. *Risk Anal.* 29, 485–487; discussion 492-497. doi: 10.1111/j.1539-6924.2008.01168.x

Kazhdan, M., Funkhouser, T., and Rusinkiewicz, S. (2003). Rotation invariant spherical harmonic representation of 3D shape descriptors. *Eurogr. Sympos. Geomet. Process.* 43, 156–165. doi: 10.2312/SGP/SGP03/156-165

Kearnes, S., McCloskey, K., Berndl, M., Pande, V., and Riley, P. (2016). Molecular graph convolutions: moving beyond fingerprints. *J. Comput. Aided Mol. Des.* 30, 595–608. doi: 10.1007/s10822-016-9938-8

Khan, P. M., and Roy, K. (2018). Current approaches for choosing feature selection and learning algorithms in quantitative structure-activity relationships (QSAR). *Expert Opin. Drug Discov.* 29, 1–15. doi: 10.1080/17460441.2018.1542428

Khened, M., Kollerathu, V. A., and Krishnamurthi, G. (2019). Fully convolutional multi-scale residual DenseNets for cardiac segmentation and automated cardiac diagnosis using ensemble of classifiers. *Med. Image Anal.* 51, 21–45. doi: 10.1016/j.media.2018.10.004

Kipf, T. N., and Welling, M. (2016). Semi-supervised classification with graph convolutional networks. *arXiv:1609.02907v4* [Preprint]. Available online at: https://arxiv.org/pdf/1609.02907.pdf

Kobayashi, K., Hashimoto, M., Honkakoski, P., and Negishi, M. (2015). Regulation of gene expression by CAR: an update. *Arch. Toxicol.* 89, 1045–1055. doi: 10.1007/s00204-015-1522-9

Krizhevsky, A., Sutskev, I., and Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. *Adv. Neural. Inf. Process. Syst.* 1, 1097–1105. doi: 10.1145/3065386

Kulkarni, A. J., Jayaraman, V. K., and Kulkarni, B. D. (2009). Review on lazy learning regressors and their applications in QSAR. *Comb. Chem. High Throughput Screen.* 12, 440–450. doi: 10.2174/138620709788167908

Kumar, A., and Chauhan, S. (2018). Use of Simplified Molecular Input Line Entry System and molecular graph based descriptors in prediction and design of pancreatic lipase inhibitors. *Future Med. Chem.* 10, 1603–1622. doi: 10.4155/fmc-2018-0024

Kuzminykh, D., Polykovskiy, D., Kadurin, A., Zhebrak, A., Baskov, I., Nikolenko, S., et al. (2018). 3D molecular representations based on the wave transform for convolutional neural networks. *Mol. Pharm.* 15, 4378–4385. doi: 10.1021/acs.molpharmaceut.7b01134

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature* 521, 436–444. doi: 10.1038/nature14539

Leist, M., Ghallab, A., Graepel, R., Marchan, R., Hassan, R., et al. (2017). Adverse outcome pathways: opportunities, limitations and open questions. *Arch. Toxicol.* 91, 3477–3505. doi: 10.1007/s00204-017-2045-3

Li, A. X., Sun, M., and Li, X. (2017). Withaferin-A induces apoptosis in osteosarcoma U2OS cell line via generation of ROS and disruption of mitochondrial membrane potential. *Eur. Rev. Med. Pharmacol. Sci.* 21, 1368–1374. doi: 10.4103/0973-1296.211042

Li, C., Cui, Z., Zheng, W., Xu, C., Ji, R., and Yang, J. (2018). Action-attending graphic neural network. *IEEE Trans. Image Process.* 27, 3657–3670. doi: 10.1109/TIP.2018.2815744

Li, H., Gong, X. J., Yu, H., and Zhou, C. (2018). Deep neural network based predictions of protein interactions using primary sequences. *Molecules* 23:E1923. doi: 10.3390/molecules23081923

Li, S., Jiang, H., and Pang, W. (2017). Joint multiple fully connected convolutional neural network with extreme learning machine for hepatocellular carcinoma nuclei grading. *Comput. Biol. Med.* 1, 156–167. doi: 10.1016/j.compbiomed.2017.03.017

Linden, A. (2006). Measuring diagnostic and predictive accuracy in disease management: an introduction to receiver operating characteristic (ROC) analysis. *J. Eval. Clin. Pract.* 12, 132–139. doi: 10.1111/j.1365-2753.2005.00598.x

Liu, K., Sun, X., Jia, L., Ma, J., Xing, H., Wu, J., et al. (2018). Chemi-Net: a molecular graph convolutional network for accurate drug property prediction. *arXiv:1803.06236v2* [Preprint]. Available online at: https://arxiv.org/abs/1803.06236

Liu, Q., Wang, Q., Xu, C., Shao, W., Zhang, C., Liu, H., et al. (2017). Organochloride pesticides impaired mitochondrial function in hepatocytes and aggravated disorders of fatty acid metabolism. *Sci. Rep.* 7:46339. doi: 10.1038/srep46339

Liu, R., Wang, H., Glover, K. P., Feasel, M. G., and Wallqvist, A. (2018). Dissecting machine-learning prediction of molecular activity: is an applicability domain needed for quantitative structure-activity relationship models based on deep neural networks? *J. Chem. Inf. Model.* 59, 117–126. doi: 10.1021/acs.jcim.8b00348

Lo, Y. C., Rensi, S. E., Torng, W., and Altman, R. B. (2018). Machine learning in chemoinformatics and drug discovery. *Drug Discov. Today* 23, 1538–1546. doi: 10.1016/j.drudis.2018.05.010

Lu, N., Wu, Y., Feng, L., and Song, J. (2018). Deep learning for fall detection: 3D-CNN combined with LSTM on video kinematic data. *IEEE J. Biomed. Health Inform.* 23, 314–323. doi: 10.1109/JBHI.2018.2808281

Luechtefeld, T., Rowlands, C., and Hartung, T. (2018). Big-data and machine learning to revamp computational toxicology and its use in risk assessment. *Toxicol. Res.* 7, 732–744. doi: 10.1039/c8tx00051d

Lumini, A., and Nanni, L. (2018). Convolutional neural networks for ATC classification. *Curr. Pharm. Des.* 24, 4007–4012. doi: 10.2174/1381612824666181112113438

Ma, J., Sheridan, R. P., Liaw, A., Dahl, G. E., and Svetnik, V. (2015). Deep neural nets as a method for quantitative structure-activity relationships. *J. Chem. Inf. Model.* 55, 263–274. doi: 10.1021/ci500747n

Mahendran, A., and Vedaldi, A. (2014). Understanding deep image representations by inverting them. *arXiv:1412.0035v1* [Preprint]. Available online at: https://arxiv.org/pdf/1412.0035.pdf

Mallat, S. (2016). Understanding deep convolutional networks. *Philos. Trans. A Math. Phys. Eng. Sci.* 374:20150203. doi: 10.1098/rsta.2015.0203

Malloy, T., Zaunbrecher, V., Beryt, E., Judson, R., Tice, R., Allard, P., et al. (2017). Advancing alternatives analysis: the role of predictive toxicology in selecting safer chemical products and processes. *Integr. Environ. Assess. Manag.* 13, 915–925. doi: 10.1002/ieam.1923

Manallack, D. T., Burden, F. R., and Winkler, D. A. (2010). Modelling inhalational anaesthetics using bayesian feature selection and QSAR modelling methods. *ChemMedChem.* 5, 1318–1323. doi: 10.1002/cmdc.201000056

Manibusan, M. K., and Touart, L. W. (2017). A comprehensive review of regulatory test methods for endocrine adverse health effects. *Crit. Rev. Toxicol.* 47, 433–481. doi: 10.1080/10408444.2016.1272095

Martinez, M., and Stiefelhagen, R. (2018). Taming the cross entropy loss. *arXiv:1810.05075v1* [Preprint]. Available online at: https://arxiv.org/pdf/1810.05075.pdf

Marty, M. S., Borgert, C., Coady, K., Green, R., Levine, S. L., Mihaich, E., et al. (2018). Distinguishing between endocrine disruption and non-specific effects on endocrine systems. *Regul. Toxicol. Pharmacol.* 99, 142–158. doi: 10.1016/j.yrtph.2018.09.002

Maturana, D., and Scherer, S. (2015). 3D Convolutional Neural Networks for landing zone detection from LiDAR. *IEEE Int. Conf. Robot. Autom.* 2015, 1050–4729. doi: 10.1109/ICRA.2015.7139679

Mayr, A., Klambauer, G., Unterthiner, T., and Hochreiter, S. (2016). DeepTox: toxicity prediction using deep learning. *Front. Environ. Sci.* 3:80. doi: 10.3389/fenvs.2015.00080

Mayr, A., Klambauer, G., Unterthiner, T., Steijaert, M., Wegner, J. K., Ceulemans, H., et al. (2018). Large-scale comparison of machine learning methods for drug target prediction on ChEMBL. *Chem. Sci.* 9, 5441–5451. doi: 10.1039/c8sc00148k

McMahon, M., Ding, S., Acosta-Jimenez, L. P., Terranova, R., Gerard, M. A., and Vitobello, A., et al. (2018). Constitutive androstane receptor 1 is constitutively bound to chromatin and 'primed' for transactivation in hepatocytes. *Mol. Pharmacol.* 95, 97–105. doi: 10.1124/mol.118.113555

Merrick, B. A., Paules, R. S., and Tice, R. R. (2015). Intersection of toxicogenomics and high throughput screening in the Tox21 program: an NIEHS perspective. *Int. J. Biotechnol.* 14, 7–27. doi: 10.1504/IJBT.2015.074797

Midzak, A. S., Chen, H., Aon, M. A., Papadopoulos, V., and Zirkin, B. R. (2011). ATP synthesis, mitochondrial function, and steroid biosynthesis in rodent primary and tumor Leydig cells. *Biol. Reprod.* 84, 976–985. doi: 10.1095/biolreprod.110.087460

Mortensen, H. M., Chamberlin, J., Joubert, B., Angrish, M., Sipes, N., Lee, J. S., et al. (2018). Leveraging human genetic and adverse outcome pathway (AOP) data to inform susceptibility in human health risk assessment. *Mamm. Genome.* 29, 190–204. doi: 10.1007/s00335-018-9738-7

Nair, V., and Hinton, G. E. (2010). "Rectified linear units improve restricted Boltzmann machines," in *Proceedings of the 27th International Conference on Machine Learning* (Haifa), 807–814.

Nguyen, Q., Mukkamala, M. C., and Hein, M. (2018). On the loss landscape of a class of deep neural networks with no bad local valleys. *arXiv:arXiv:1809.10749v1* [Preprint]. Available online at: https://arxiv.org/abs/1809.10749

Nikolic, K., Filipic, S., and Agbaba, D. (2012). Multi-target QSAR and docking study of steroids binding to corticosteroid-binding globulin and sex hormone-binding globulin. *Curr. Comput. Aided Drug Des.* 8, 296–308. doi: 10.2174/157340912803519642

NRC. (2007). *Toxicity Testing in the 21st Century: A Vision and a Strategy*. Washington, DC: The National Academies Press.

Nusrat, S., and Khan, R. H. (2018). Exploration of ligand-induced protein conformational alteration, aggregate formation, and its inhibition: a biophysical insight. *Prep. Biochem. Biotechnol.* 48, 43–56. doi: 10.1080/10826068.2017.1387561

Pastur-Romay, L. A., Cedron, F., Pazos, A., and Porto-Pazos, A. B. (2016). Deep artificial neural networks and neuromorphic chips for big data analysis: pharmaceutical and bioinformatics applications. *Int. J. Mol. Sci.* 17:1313. doi: 10.3390/ijms17081313

Peng, Y., Rios, A., Kavuluru, R., and Lu, Z. (2018). Extracting chemical-protein relations with ensembles of SVM and deep learning models. *Database*. 2018:bay073. doi: 10.1093/database/bay073

Pham, T., Tran, T., Phung, D., and Venkatesh, S. (2017). Predicting healthcare trajectories from medical records: a deep learning approach. *J. Biomed. Inform.* 69, 218–229. doi: 10.1016/j.jbi.2017.04.001

Philbrick, K. A., Yoshida, K., Inoue, D., Akkus, Z., Kline, T. L., Weston, A. D., et al. (2018). What does deep learning see? Insights from a classifier trained to predict contrast enhancement phase from CT images. *AJR Am. J. Roentgenol.* 211, 1184–1193. doi: 10.2214/AJR.18.20331

Poernomo, A., and Kang, D. K. (2018). Biased dropout and crossmap dropout: learning towards effective dropout regularization in convolutional neural network. *Neural. Netw.* 104, 60–67. doi: 10.1016/j.neunet.2018.03.016

Polishchuk, P. (2017). Interpretation of quantitative structure-activity relationship models: past, present, and future. *J. Chem. Inf. Model.* 57, 2618–2639. doi: 10.1021/acs.jcim.7b00274

Pu, J., and Ramani, K. (2006). On visual similarity based 2D drawing retrieval. *Computer-Aided Design.* 38, 249–259. doi: 10.1016/j.cad.2005.10.009

Putz, M. V., and Dudaş, N. A. (2013). Determining chemical reactivity driving biological activity from SMILES transformations: the bonding mechanism of anti-HIV pyrimidines. *Molecules.* 18, 9061–9116. doi: 10.3390/molecules18089061

Qatanani, M., and Moore, D. D. (2005). CAR, the continuously advancing receptor, in drug metabolism and disease. *Curr. Drug Metab.* 6, 329–339. doi: 10.2174/1389200054633899

Qiao, H., Wu, J., Li, X., Shoreh, M. H., Fan, J., and Dai, Q. (2018). GPU-based deep convolutional neural network for tomographic phase microscopy with l1 fitting and regularization. *J. Biomed. Opt.* 23, 1–7. doi: 10.1117/1.JBO.23.6.066003

Qiu, Y., Yan, S., Gundreddy, R. R., Wang, Y., Cheng, S., Liu, H., et al. (2017). A new approach to develop computer-aided diagnosis scheme of breast mass classification using deep learning technology. *J. Xray Sci. Technol.* 25, 751–763. doi: 10.3233/XST-16226

Ragoza, M., Hochuli, J., Idrobo, E., Sunseri, J., and Koes, D. R. (2017). Protein-ligand scoring with convolutional neural networks. *J. Chem. Inf. Model.* 57, 942–957. doi: 10.1021/acs.jcim.6b00740

Rawat, W., and Wang, Z. (2017). Deep convolutional neural networks for image classification: a comprehensive review. *Neural. Comput.* 29, 2352–2449. doi: 10.1162/NECO_a_00990

Reitz, M., Sacher, O., Tarkhov, A., Trumbach, D., and Gasteiger, J. (2004). Enabling the exploration of biochemical pathways. *Org. Biomol. Chem.* 2, 3226–3237. doi: 10.1039/B410949J

Renner, S., Schwab, C. H., Gasteiger, J., and Schneider, G. (2006). Impact of conformational flexibility on three-dimensional similarity searching using correlation vectors. *J. Chem. Inf. Model.* 46, 2324–2332. doi: 10.1021/ci050075s

Richard, A. M., Judson, R. S., Houck, K. A., Grulke, C. M., Volarath, P., Thillainadarajah, I., et al. (2016). ToxCast Chemical landscape: paving the road to 21st century toxicology. *Chem. Res. Toxicol.* 29, 1225–1251. doi: 10.1021/acs.chemrestox.6b00135

Rogers, D., and Hahn, M. (2010). Extended-connectivity fingerprints. *J. Chem. Inf. Model.* 50, 742–754. doi: 10.1021/ci100050t

Roy, K., and Das, R. N. (2014). A review on principles, theory and practices of 2D-QSAR. *Curr. Drug Metab.* 15, 346–379. doi: 10.2174/1389200215666140908102230

Sadowski, J., Gasteiger, J., and Klebe, G. (1994). Comparison of automatic three-dimensional model builders using 639 X-ray structures. *J. Chem. Inf. Comput. Sci.* 34, 1000–1008. doi: 10.1021/ci00020a039

Saha, M., Chakraborty, C., and Racoceanu, D. (2018). Efficient deep learning model for mitosis detection using breast histopathology images. *Comput. Med. Imaging Graph.* 64, 29–40. doi: 10.1016/j.compmedimag.2017

Salmaso, V., and Moro, S. (2018). Bridging molecular docking to molecular dynamics in exploring ligand-protein recognition process: an overview. *Front. Pharmacol.* 9:923. doi: 10.3389/fphar.2018.00923

Sato, M., Horie, K., Hara, A., Miyamoto, Y., Kurihara, K., Tomio, K., et al. (2018). Application of deep learning to the classification of images from colposcopy. *Oncol. Lett.* 15, 3518–3523. doi: 10.3892/ol.2018.7762

Sayers, E. W., Agarwala, R., Bolton, E. E., Brister, J. R., Canese, K., Clark, K., et al. (2018). Database resources of the National center for biotechnology information. *Nucleic Acids Res.* 33, D39–D45. doi: 10.1093/nar/gky1069

Scalfani, V. F., Williams, A. J., Tkachenko, V., Karapetyan, K., Pshenichnov, A., Hanson, R. M., et al. (2016). Programmatic conversion of crystal structures into 3D printable files using Jmol. *J. Cheminform.* 8:66. doi: 10.1186/s13321-016-0181-z

Schlichtkrull, M., Kipf, T. N., Bloem, P., van den Berg, R., Titov, I., and Welling, M. (2017). Modeling relational data with graph convolutional networks. *arXiv:1703.06103v4* [Preprint]. Available online at: https://arxiv.org/pdf/1703.06103.pdf

Schwab, C. H. (2010). Conformations and 3D pharmacophore searching. *Drug Discovery Today Technol.* 7, e245–e253. doi: 10.1016/j.ddtec.2010.10.003

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. (2016). Grad-CAM: visual explanations from deep networks via gradient-based localization. *arXiv:1610.02391v3* [Preprint]. Available online at: https://arxiv.org/abs/1610.02391

Shaughnessy, D. T., McAllister, K., Worth, L., Haugen, A. C., Meyer, J. N., Domann, F. E., et al. (2014). Mitochondria, energetics, epigenetics, and cellular responses to stress. *Environ. Health Perspect*. 122, 1271–1278. doi: 10.1289/ehp.1408418

Shen, X., Tian, X., Liu, T., Xu, F., and Tao, D. (2018). Continuous dropout. *IEEE Trans. Neural. Netw. Learn Syst.* 29, 3926–3937. doi: 10.1109/TNNLS.2017.2750679

Sifakis, S., Androutsopoulos, V. P., Tsatsakis, A. M., and Spandidos, D. A. (2017). Human exposure to endocrine disrupting chemicals: effects on the male and female reproductive systems. *Environ. Toxicol. Pharmacol.* 51, 56–70. doi: 10.1016/j.etap.2017.02.024

Silva, F. T., and Trossini, G. H. (2014). The survey of the use of QSAR methods to determine intestinal absorption and oral bioavailability during drug design. *Med. Chem.* 10, 441–448. doi: 10.2174/15734064106661404151221 15

Simões, R. S., Maltarollo, V. G., Oliveira, P. R., and Honorio, K. M. (2018). Transfer and multi-task learning in QSAR modeling: advances and challenges. *Front. Pharmacol.* 9:74. doi: 10.3389/fphar.2018.00074

Simonyan, K., Vedaldi, A., and Zisserman,. A. (2013). Deep inside convolutional networks: visualising image classification models and saliency maps. *arXiv:1312.6034v2* [Preprint]. Available online at: https://arxiv.org/pdf/1312.6034.pdf

Sipes, N. S., Wambaugh, J. F., Pearce, R., Auerbach, S. S., Wetmore, B. A., Hsieh, J. H., et al. (2017). An intuitive approach for predicting potential human health risk with the Tox21 10k library. *Environ. Sci. Technol.* 51, 10786–10796. doi: 10.1021/acs.est.7b00650

Smilkov, D., Thorat, N., Kim, B., Viegas, F., and Wattenberg, M. (2017). SmoothGrad: removing noise by adding noise. *arXiv:1706.03825v1* [Preprint]. Available online at: https://arxiv.org/pdf/1706.03825.pdf

Steven, E. O., and Han, D. S. (2018). Feature representation and data augmentation for human activity classification based on wearable IMU sensor data using a deep LSTM neural network. *Sensors* 18:E2892. doi: 10.3390/s18092892

Suárez-Paniagua, V., and Segura-Bedmar, I. (2018). Evaluation of pooling operations in convolutional architectures for drug-drug interaction extraction. *BMC Bioinformatics* 19:209. doi: 10.1186/s12859-018-2195-1

Szegedy, C., Liue, W., Jiam, Y., Sermanet, P., Reed, S., Anguelov, D., et al. (2014). Going deeper with convolutions. *arXiv:1409:4842v1* [Preprint]. Available online at: https://arxiv.org/abs/1409.4842v1

Talevi, A., Bellera, C. L., Di Ianni, M., Duchowicz, P. R., Bruno-Blanch, L. E., and Castro, E. A. (2012). An integrated drug development approach applying topological descriptors. *Curr. Comput. Aided Drug Des.* 8, 172–181. doi: 10.2174/157340912801619076

Tapia-Orozco, N., Santiago-Toledo, G., Barrón, V., Espinosa-García, A. M., García-García, J. A., and García-Arrazola, R. (2017). Environmental epigenomics: current approaches to assess epigenetic effects of endocrine disrupting compounds (EDC's) on human health. *Environ. Toxicol. Pharmacol.* 51, 94–99. doi: 10.1016/j.etap.2017.02.004

Tetko, I. V., Gasteiger, J., Todeschini, R., Mauri, A., Livingstone, D., Ertl, P., et al. (2005). Virtual computational chemistry laboratory—design and description. *J. Comput. Aided Mol. Des.* 19, 453–463. doi: 10.1007/s10822-005-8694-y

Tice, R. R., Austin, C. P., Kavlock, R. J., and Bucher, J. R. (2013). Improving the human hazard characterization of chemicals: a Tox21 update. *Environ. Health Perspect.* 121, 756–765. doi: 10.1289/ehp.1205784

Tollefsen, K. E., Scholz, S., Cronin, M. T., Edwards, S. W., de Knecht, J., Crofton, K., et al. (2014). Applying adverse outcome pathways (AOPs) to support integrated approaches to testing and assessment (IATA). *Regul. Toxicol. Pharmacol.* 70, 629–640. doi: 10.1016/j.yrtph.2014.09.009

Tox21 Data Challenge. (2014). *National Center for Advancing Translational Sciences*. Available online at: https://tripod.nih.gov/tox21/challenge/

Tuffery, P., and Derreumaux, P. (2017). Flexibility and binding affinity in protein-ligand, protein-protein and multi-component protein interactions: limitations of current computational approaches. *J. R. Soc. Interface* 9, 20–33. doi: 10.1098/rsif.2011.0584

Tustison, N. J., Avants, B. B., Lin, Z., Feng, X., Cullen, N., Mata, J. F., et al. (2018). Convolutional neural networks with template-based data augmentation for functional lung image quantification. *Acad. Radiol.* 5:e3. doi: 10.1016/j.acra.2018.08.003

Uesawa, Y. (2018). Quantitative structure–activity relationship analysis using deep learning based on a novel molecular image input technique. *Bioorg. Med. Chem. Lett.* 28, 3400–3403. doi: 10.1016/j.bmcl.2018.08.032

Vakli, P., Deák-Meszlényi, R. J., Hermann, P., and Vidnyánszky, Z. (2018). Transfer learning improves resting-state functional connectivity pattern analysis using convolutional neural networks. *Gigascience* 7:e130. doi: 10.1093/gigascience/giy130

Voulodimos, A., Doulamis, N., Doulamis, A., and Protopapadakis, E. (2018). Deep learning for computer vision: a brief review. *Comput. Intell. Neurosci.* 2018:7068349. doi: 10.1155/2018/7068349

Wang, C., Chen, H., and Ying, W. (2018). Cytosolic aspartate aminotransferase mediates the mitochondrial membrane potential and cell survival by maintaining the calcium homeostasis of BV2 microglia. *Neuroreport.* 29, 99–105. doi: 10.1097/WNR.0000000000000914

Wang, S. H., Phillips, P., Sui, Y., Liu, B., Yang, M., and Cheng, H. (2018). Classification of Alzheimer's disease based on eight-layer convolutional neural network with leaky rectified linear unit and max pooling. *J. Med. Syst.* 42:85. doi: 10.1007/s10916-018-0932-7

Wang, Y., Xiao, J., Suzek, T. O., Zhang, J., Wang, J., and Bryant, S. H. (2009). PubChem: a public information system for analyzing bioactivities of small molecules. *Nucleic Acids Res.* 37, W623–W633. doi: 10.1093/nar/gkp456

Wang, Y., Xing, J., Xu, Y., Zhou, N., Peng, J., Xiong, Z., et al. (2015). *In silico* ADME/T modelling for rational drug design. *Q. Rev. Biophys.* 48, 488–515. doi: 10.1017/S0033583515000190

Weininger, D. (1988). SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *J. Chem. Inf. Comput. Sci.* 28, 31–36. doi: 10.1021/ci00057a005

Xia, M., Huang, R., Shi, Q., Boyd, W. A., Zhao, J., Sun, N., et al. (2018). Comprehensive analyses and prioritization of Tox21 10K chemicals affecting mitochondrial function by in-depth mechanistic studies. *Environ. Health Perspect.* 126:077010. doi: 10.1289/EHP2589

Xiang, Y., Mottaghi, R., and Savarese, S. (2014). "Beyond PASCAL: a benchmark for 3D object detection in the wild," in *IEEE Winter Conference on Applications of Computer Vision* (Steamboat Springs). doi: 10.1109/WACV.2014.6836101

Xu, Y., Pei, J., and Lai, L. (2017). Deep learning based regression and multiclass models for acute oral toxicity prediction with automatic chemical feature extraction. *J. Chem. Inf. Model.* 57, 2672–2685. doi: 10.1021/acs.jcim.7b00244

Yang, L., Zhang, J., Che, X., and Gao, Y. Q. (2016). Simulation studies of protein and small molecule interactions and reaction. *Methods Enzymol.* 578, 169–212. doi: 10.1016/bs.mie.2016.05.031

Yap, C. W., Li, H., Ji, Z. L., and Chen, Y. Z. (2007). Regression methods for developing QSAR and QSPR models to predict compounds of specific pharmacodynamic, pharmacokinetic and toxicological properties. *Mini Rev. Med. Chem.* 7, 1097–1107. doi: 10.2174/138955707782331696

Yun, X., Rao, W., Xiao, C., and Huang, Q. (2017). Apoptosis of leukemia K562 and Molt-4 cells induced by emamectin benzoate involving mitochondrial membrane potential loss and intracellular Ca^{2+} modulation. *Environ. Toxicol. Pharmacol.* 52, 280–287. doi: 10.1016/j.etap.2017.04.013

Zaharescu, A., Boyer, E., Varanasi, K., and Horaud, R. (2009). "Surface feature detection and description with applications to mesh matching," in *2009 IEEE Conference on Computer Vision and Pattern Recognition* (Florida) 373–380. doi: 10.1109/CVPR.2009.5206748

Zhang, L., Tan, J., Han, D., and Zhu, H. (2017). From machine learning to deep learning: progress in machine intelligence for rational drug discovery. *Drug Discov. Today* 22, 1680–1685. doi: 10.1016/j.drudis.2017.08.010

Zhang, Q., Li, J., Middleton, A., Bhattacharya, S., and Conolly, R. B. (2018). Bridging the data gap from *in vitro* toxicity testing to chemical safety assessment through computational modeling. *Front. Public Health* 6:261. doi: 10.3389/fpubh.2018.00261

Zhang, Z., and Sabuncu, M. R. (2018). Generalized cross entropy loss for training deep neural networks with noisy labels. *arXiv:1805.07836v4*. Available online at: https://arxiv.org/pdf/1805.07836.pdf

Zhen, X., Chen, J., Zhong, Z., Hrycushko, B., Zhou, L., Jiang, S., et al. (2017). Deep convolutional neural network with transfer learning for rectum toxicity prediction in cervical cancer radiotherapy: a feasibility study. *Phys. Med. Biol.* 62, 8246–8263. doi: 10.1088/1361-6560/aa8d09

Zhou, Z., and Li, X. (2017). Convolution on graph: a high-orderand adaptive approach. *arXiv:1706.09916v2* [Preprint]. Available online at: https://arxiv.org/pdf/1706.09916.pdf

Zhu, H., Zhang, J., Kim, M. T., Boison, A., Sedykh, A., and Moran, K. (2014). Big data in chemical toxicity research: the use of high-throughput screening assays to identify potential toxicants. *Chem. Res. Toxicol.* 27, 1643–1651. doi: 10.1021/tx500145h

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

frontiers in Genetics

ORIGINAL RESEARCH
published: 24 April 2019
doi: 10.3389/fgene.2019.00351

A Novel Deep Neural Network Model for Multi-Label Chronic Disease Prediction

*Xiaoqing Zhang, Hongling Zhao, Shuo Zhang and Runzhi Li**

Collaborative Innovation Center of Internet Healthcare, Zhengzhou University, Zhengzhou, China

Chronic diseases are one of the biggest threats to human life. It is clinically significant to predict the chronic disease prior to diagnosis time and take effective therapy as early as possible. In this work, we use problem transform methods to convert the chronic diseases prediction into a multi-label classification problem and propose a novel convolutional neural network (CNN) architecture named GroupNet to solve the multi-label chronic disease classification problem. Binary Relevance (BR) and Label Powerset (LP) methods are adopted to transform multiple chronic disease labels. We present the correlated loss as the loss function used in the GroupNet, which integrates the correlation coefficient between different diseases. The experiments are conducted on the physical examination datasets collected from a local medical center. In the experiments, we compare GroupNet with other methods and models. GroupNet outperforms others and achieves the best accuracy of 81.13%.

Keywords: multi-label classification, chronic disease, group block, GroupNet, correlated loss

OPEN ACCESS

Edited by:
Minjun Chen,
National Center for Toxicological Research (FDA), United States

Reviewed by:
Harinder Singh,
J. Craig Venter Institute, United States
Xishuang Dong,
Prairie View A&M University, United States
Mingzhen Zhang,
National Cancer Institute (NIH), United States

****Correspondence:***
Runzhi Li
rzli@ha.edu.cn

Specialty section:
This article was submitted to Bioinformatics and Computational Biology, a section of the journal Frontiers in Genetics

Received: *29 September 2018*
Accepted: *01 April 2019*
Published: *24 April 2019*

Citation:
Zhang X, Zhao H, Zhang S and Li R (2019) A Novel Deep Neural Network Model for Multi-Label Chronic Disease Prediction. Front. Genet. 10:351. doi: 10.3389/fgene.2019.00351

INTRODUCTION

Chronic diseases account for a majority of healthcare costs and they have been the main cause of mortality in the worldwide (Lehnert et al., 2011; Shanthi et al., 2015). With the development of preventive medicine, it is very important to predict chronic diseases as early as possible. However, it is difficult for clinicians to make useful diagnosis in advance, because the pathogeny of chronic disease is fugacious and complex. In general, clinicians firstly form the diagnostic results of chronic disease according to the physical examination records based on their expertise and experience. Nevertheless, with more and more physical examination records produced, clinicians would have difficulty forming accurate diagnosis in limited time. Artificial intelligence technology has brought enormous reform in medical domain, and it can help doctor diagnose by forming the diagnostic results automatically based on the prediction models. In clinical practice, a symptom is always associated with multiple chronic diseases based on the physical examination records. Hence, the diagnosis or prediction of multiple chronic diseases could be transformed into a multi-label classification problem.

Multi-label classification problem is one of the supervised learning problems where an instance may be associated with multiple labels simultaneously. Currently, Multi-label classification problems have appeared in more and more applications, such as diseases prediction, semantic analysis, object tracking, and image classification, etc. Many successful multi-label algorithms have been obtained by the problem transformation methods. Problem transformation methods firstly convert the multi-label classification problems into several binary classification problems or a multi-class classification problem, and then apply original machine learning algorithms to handle them. The binary relevance (BR) method and label powerset (LP) method (Zhang and Zhou, 2014) are two representative label transformation methods. Plenty of competitive machine learning algorithms have been proposed based on problem transformation methods in the literatures, such as support vector machines (SVM) (Gu et al., 2015; Khan et al., 2018), decision tree (DT) (Hong et al., 2018), random forest (RF) (Murphy, 2018), etc.

Currently, deep learning technique is applied to various fields successfully since it provides a more efficient learning mechanism for classification problems than classical machine learning methods. For medical data analysis, numerous machine learning methods have been applied to analyze various medical data. BPMLL (Zhang and Zhou, 2006) is a back-propagation neural network for multi-label functional genomics classification, and it addresses correctly predicted labels that should be ranked higher than those mistakenly predicted labels by modifying the loss function. Lipton et al. (2015) utilized the LSTM to analyze time-series clinical data to diagnose 128 different diseases. In order to reduce over-fitting and improve the classification performance of the LSTM architecture, label replication and auxiliary outputs strategies were applied in their work. Maxwell et al. (2017) used a 2-layer deep neural network to classify three chronic diseases based on physical examination records and found combine deep learning algorithms with RAkEL (Tsoumakas and Vlahavas, 2007) method that could improve multi-label classification performance. Miotto et al. (2016) combined a 3-layer autoencoder (AE) and logistic regression classifiers to predict ICD 9-based disease diagnosis using a prediction window. Liang et al. (2014) used a Deep Belief Network (DBN) to generate patient vectors, and then applied a support vector machine (SVM) to classify these generated patient vectors for general disease diagnoses. Jin et al. (2018) made hospital mortality prediction with medical named entities and multimodal learning based on the Long Short-Term Memory (LSTM) architecture, and they outperformed the benchmark by 2% AUC. However, applying deep learning technique to the medical data is still challenging because medical data are sparse, heterogeneous and unstructured.

In this work, we apply the convolutional neural network (CNN) to handle the classification of multiple chronic diseases based on the physical examination records. Because the CNN is the most widely used deep learning method, and it usually gets the desirable classification performance in various classification problems (such as medical image analysis, medical text analysis, and disease prediction). For multiple chronic diseases label transformation, we use two common problem transformation methods: binary relevance (BR) and label powerset (LP) methods in the data preprocessing phase, in order to get expected performance. BR converts multiple chronic disease classification problem into several binary chronic disease classification problems while LP transforms multiple chronic disease classification in a single-label multi-class classification problem.

The main contributions of this work can be summarized as following. Firstly, we devise the convolution block named group block, which both decreases the number of convolution parameter and enhances the overall classification performance. Secondly, a novel CNN architecture named GroupNet using group block is proposed for the classification of multiple chronic diseases based on the physical examination dataset. Thirdly, we devise the correlated loss (CL) to improve the classification performance used in the proposed GroupNet. The proposed GroupNet achieves the best accuracy of 81.13% and increases the overall classification results by at least 2.57% than any other state-of-art deep learning and machine learning methods.

The rest of this work is organized as follows. Section Dataset and Data Preprocessing introduces dataset and data preprocessing. Section Problem Formulation provides definition of the multi-label chronic disease prediction problem. The group convolution strategy, group block and GroupNet architecture are presented in Section Methods. Correlation loss and optimization strategies are elucidated in Section Loss Function and Optimization. Section Experiments and Evaluation describes experiment setup and evaluation measures. Results and Discussion are illuminated in Section Results and Discussion. Finally, Conclusions concludes this work along with future work.

DATASET AND DATA PREPROCESSING

In the work, we mainly focus on multiple chronic disease classification. It can be formulated into a multi-label classification problem. There are three common chronic diseases are selected from the physical examination records: hypertension (H), diabetes (D), and fatty liver (FL).

In the experiments, the physical examination datasets are collected from a local medical center, which contain 110,300 physical examination records from about 80,000 anonymous patients (Li et al., 2017a,b). Sixty-two feature items are selected from over 100 examination items based on medical expert experience and related literature in every physical examination record. These feature items contain 4 basic physical examination items, 26 blood routine items, 12 urine routine items, and 20 items from liver function.

Two multi-label transformation methods consisting of binary relevance (BR) and label powerset (LP) method are used in this work. For BR method, the diagnosis of a given patient can be one of three possible results: all three chronic diseases, different combination of the chronic diseases, or no signs of any three chronic diseases, which means that there are totally eight different sets of diagnoses {000, 100, 010, 001, 110, 101, 011, 111}. Based on Label Powerset (LP) method, we get eight different prediction labels and can be represented by {0, 1, 2, 3, 4, 5, 6, 7}.

In order to understand dataset better and receive expected results, we do some data analysis in the stage of data preprocessing as shown in **Figure 1**. **Figure 1A** presents the multi-label distribution of chronic diseases, and single-label distribution of three chronic diseases is shown in **Figure 1B**. The results demonstrate that the multi-label distribution of chronic diseases is highly skewed, 62.5% of physical examination records is occupied by normal and HFL, and while independent diabetes (D) only hold 1% of physical examination records according to **Figure 1A**. The single label distribution of fatty liver is

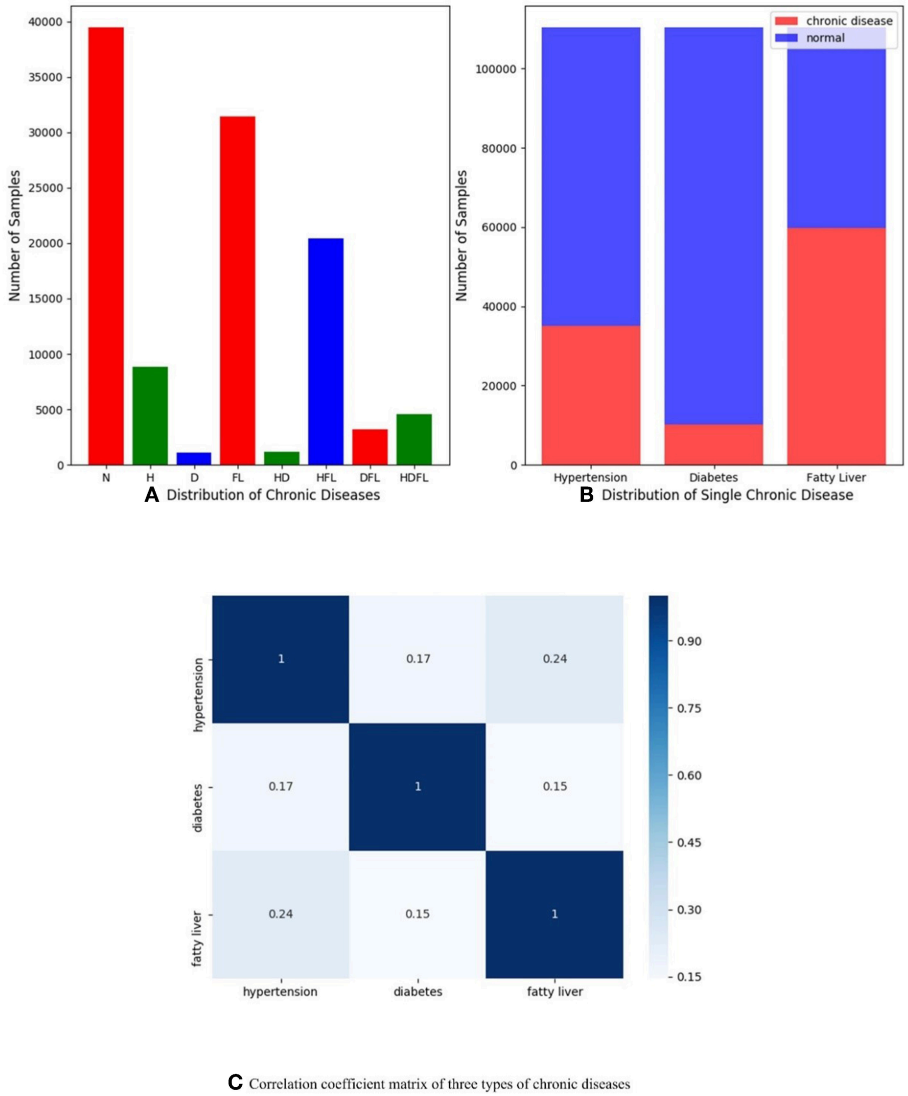


FIGURE 1 | (A) Distribution of multiple chronic diseases; **(B)** Distribution of single-label of three chronic diseases dependencies; **(C)** Correlation coefficient matrix of three types of chronic diseases (hypertension, diabetes, and fatty liver), and they are computed by Pearson product-moment correlation coefficient.

a balanced proportion, while the single label distributions of hypertension and diabetes are both imbalanced as you can see from **Figure 1B**. The correlation coefficient analysis can indicate the label dependencies, and it can be calculated by Pearson product-moment correlation coefficient (PMMC) (Mohamad Asri et al., 2018; Weber and Immink, 2018). **Figure 1C** shows that the correlation coefficient value between hypertension and flatty liver is maximum among three chronic disease pairs, but the correlation coefficient value is only 0.24. According to the theory of correlation coefficient, we can infer that the correlation between three chronic diseases are not strong.

We firstly use simple data augmentation method to handle label imbalance problem. However, this method does not work as we expected likely due to the fact that correlation coefficient value among diseases is small as you can see in **Figure 1C**. Focal loss (Lin et al., 2017) strategy is utilized to relieve label imbalance problem in this work. Furthermore, a cost-sensitive loss learning algorithm called correlated loss (CL) would be described in Group Convolution Strategy in detail and correlation coefficient values between chronic diseases is used as hyper-parameters in the correlated loss. The correlation loss is mainly proposed for improving overall classification performance. Physical examination data are split into two parts, 70% of the data for training and 30% of the data for testing in the experiments.

PROBLEM FORMULATION

In medicine filed, the goal of multiple chronic diseases prediction is to predict onset of chronic diseases in advance based on disease prediction model. To this end, we solve multiple chronic diseases prediction problem based on the physical examination dataset. It can be formulated into a multi-label classification problem in computer science. Firstly, we use problem transform methods to transform multiple chronic disease classification problem into multi-label problem classification. Secondly we construct CNN architectures to resolve the multi-label classification.

METHODS

Group Convolution Strategy

To improve the performance of a convolutional neural network (CNN) architecture. It is easy to be adopted that we increase the number of convolution kernel in every convolution layer simply. However, it would increase the number of convolution parameter and weaken the classification results. Some well-known and successful convolutional neural network architectures have been proposed to handle this problem, such as IGCNets (Zhang et al., 2017; Sun et al., 2018; Xie et al., 2018), and ShuffleNet (Ma et al., 2018). One common ground for these CNN architectures is that they are implemented based on group convolution strategy (Krizhevsky et al., 2012).

In the implementation of the group convolution strategy, there are being two continuous convolution layers at least. The number of convolution kernel in every convolution layer is split into several independent group convolution partitions. An example of group convolution strategy is shown in **Figure 2**. A CNN model consists of two continuous convolution layers, in which m and n convolution kernels are set respectively. By the group convolution strategy, we split every convolution layer into two partition convolution units and the number of convolution kernels is the half. The reduction of convolution parameters is shown in Equation 1.

$$\frac{2 \times \frac{m}{2} \times \frac{n}{2} \times 1 \times 3}{m \times n \times 1 \times 3} = \frac{1}{2} \tag{1}$$

Group Block

Inspired by group convolution strategy, we propose the group block in this work. Group block consists of two parts, which are group convolution and cluster convolution. The architecture of group block is shown in **Figure 3**.

In the group convolution part, it splits one convolution unit to multiple partition convolution units. The number of partition convolution units can be set randomly for different convolution layers L. For example, it can be set to split M or N convolution units. In cluster convolution part, a 1 × 1 convolution layer is designed after the group convolution part. It is implemented to cluster the correlated feature maps and enhances discriminability for local patches within the receptive field.

The parameters of group block are described by (L, N_i (i = 1, … m), j). Here L denotes the number of continuous convolution layers. N_i (i = 1, … m) shows the number of partition convolution units in the ith convolutional layer. j is the number of cluster convolution layers.

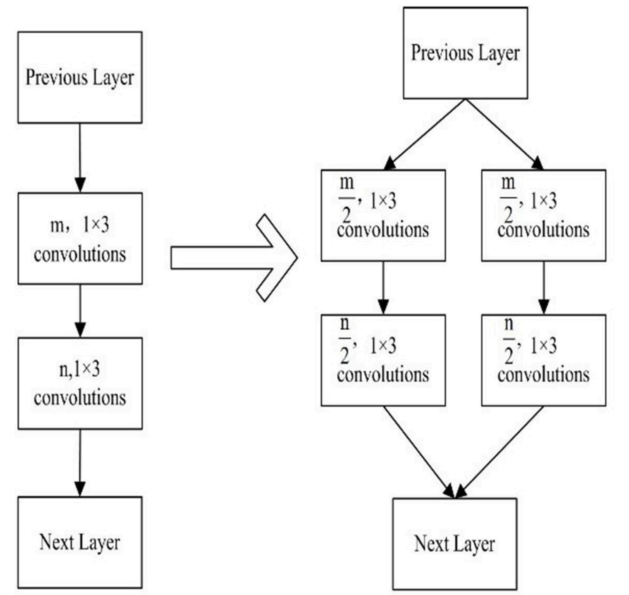


FIGURE 2 | Group convolution strategy.

GroupNet Architecture

In this work, we construct the CNN architecture based on the proposed group block named the GroupNet, shown in **Figure 4**. The proposed group block is the core part of the GroupNet, which is a variant of group convolution. The main difference between the proposed group block and the traditional group convolution is that we add a cluster convolution part after group convolution part in group block. Hence, the GroupNet architecture built on the group block improves the classification performance efficiently when comparing to several advanced CNN architectures.

The GroupNet architecture contains six layers: input layer, group block, max-pooling layer, dropout layer, fully-connected layer and softmax layer. The detail parameters of GroupNet architecture is listed in **Figure 5**. Small convolution kernels always are used to reduce the computation burden and improve the classification performance (Huang et al., 2016; Iandola et al., 2016; Sandler et al., 2018). In this work, we use 1 × 3 as the convolution kernel size. Because convolution kernel size 1 × 3 achieves better performance than other convolution kernel sizes in the experiments. Because physical examination data are one-dimensional. Hence, one-dimensional convolution kernel is adopted. Furthermore, softmax function is used as classifier, because it is standard to use the softmax as classifier in deep learning.

Well-known dropout (Srivastava et al., 2014; Bouthillier et al., 2015) technique is available to alleviate over-fitting for CNN. In this work, we set a dropout layer between the max-pooling layer and the fully-connected layer and the drop rate is 0.5 which is set experimentally.

In this work, LP and BP are adopted to resolve the multi-label classification, respectively. LP method is to transform

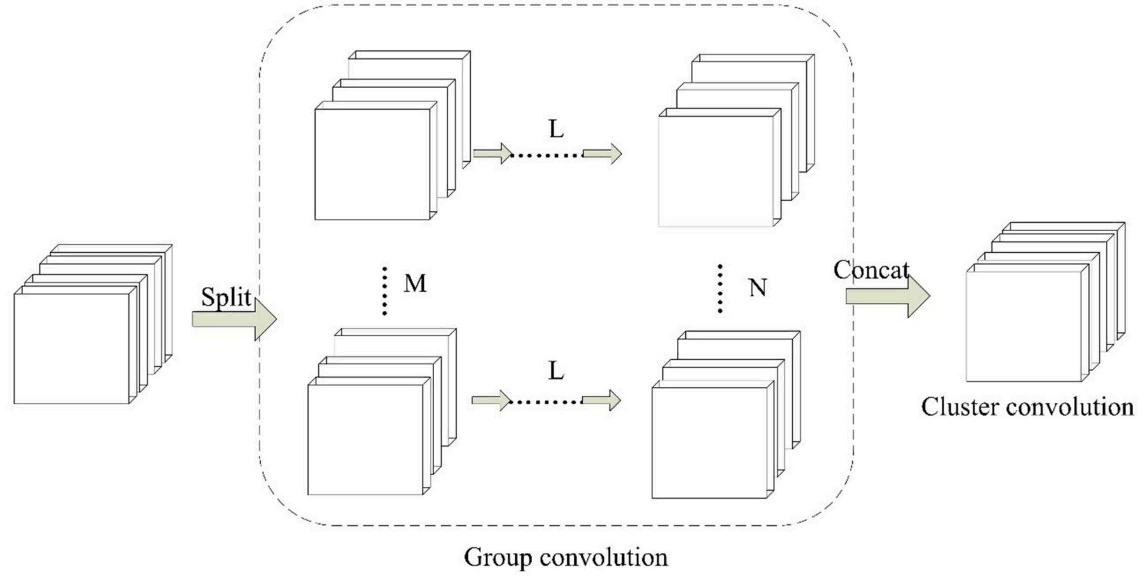
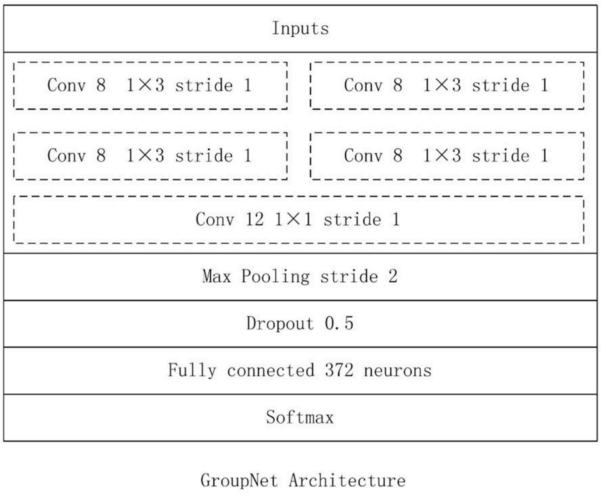


FIGURE 3 | The paradigm of Group Block. For L continuous convolution layers, M and N denotes the number of independent partition convolution units.


Inputs

Conv 8 1×3 stride 1 | Conv 8 1×3 stride 1

Conv 8 1×3 stride 1 | Conv 8 1×3 stride 1

Conv 12 1×1 stride 1

Max Pooling stride 2

Dropout 0.5

Fully connected 372 neurons

Softmax

GroupNet Architecture

FIGURE 4 | GroupNet Architecture.

multiple chronic disease classification into the single-label multi-class classification, while BR method converts the multi-label chronic disease classification into three binary classifications. Correspondingly, LP-GroupNet and BR-GroupNet are named in experiments.

LOSS FUNCTION AND OPTIMIZATION

Correlated Loss

Focal loss (FL) (Lin et al., 2017) is a variant of standard cross entropy loss, and it alleviates loss of correctly classified examples domain the gradient in the training and can be computed as following.

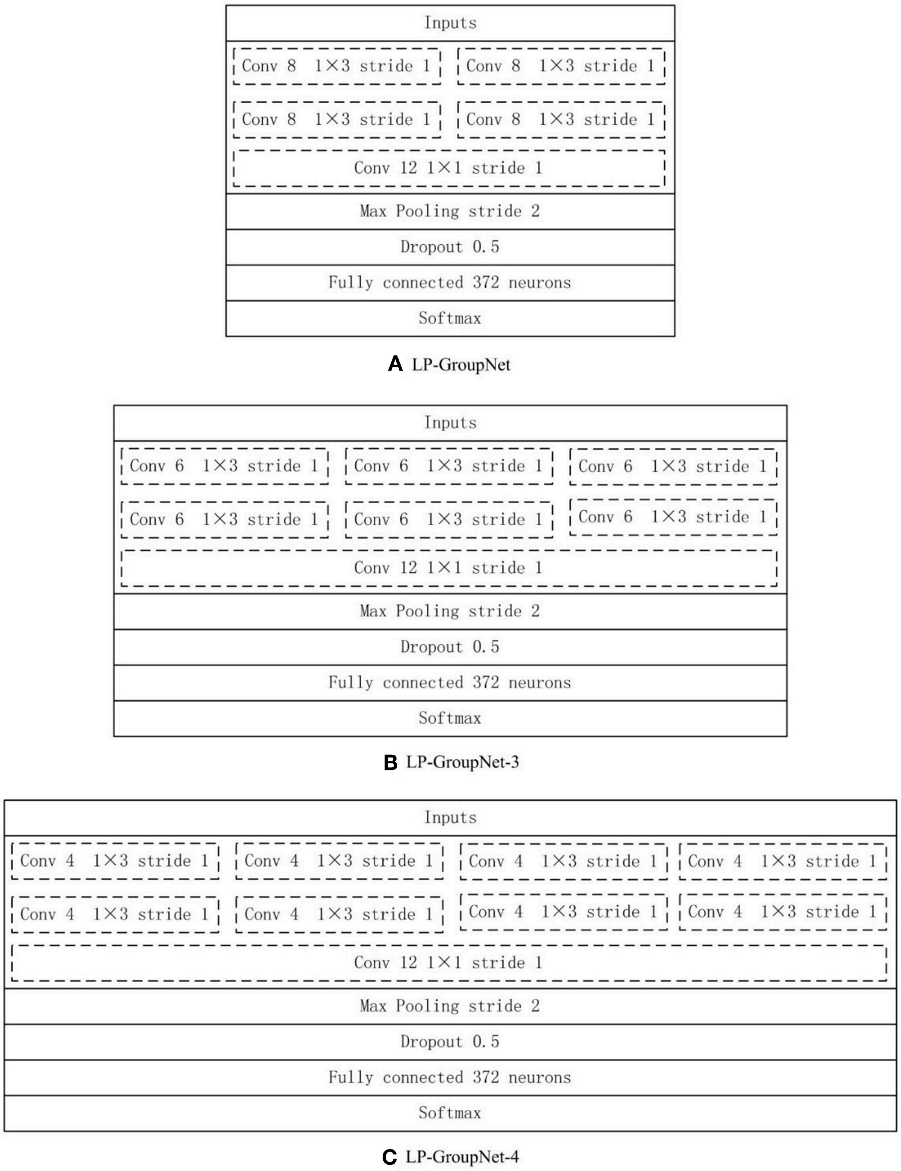
$$FL(p) = -(1-p)^{\gamma} \log p \tag{2}$$

Here p is the probability for predicted label. $(1-p)^{\gamma}$ is modulating factor and γ is a focusing parameter. γ is set manually. When $\gamma = 0$, focal loss is equal to standard cross entropy loss. The cross entropy loss is described as following.

$$CE(p, q) = -q \log p \tag{3}$$

CE (p, q) is a cross entropy loss, p and q represent the expected output and actual output, respectively.

In the BR-GroupNet architecture, each binary classifier is independent of each other, in order to enhance connection between independent classifiers and each classifier can learn useful information from each other. Hence, we propose a cost-sensitive learning algorithm named correlated loss (CL) for the BR-GroupNet to enhance classification performance by learning loss information from each other. In the BR-GroupNet architecture, the correlated loss of each binary classifier consists of two parts: main loss and auxiliary loss. Main loss can be computed by the classifier itself and auxiliary loss is the sum of product associated classifier loss and correlation coefficient value. In this work, correlation coefficient value between two chronic diseases is chosen as a hyper parameter in auxiliary loss, because correlation coefficient value between two diseases is small and it also indicates disease dependencies between two diseases. Therefore, correlated loss (CL) of an



A LP-GroupNet

B LP-GroupNet-3

C LP-GroupNet-4

FIGURE 5 | (A) LP-GroupNet, **(B)** LP-GroupNet-3, **(C)** LP-GroupNet-4.

independent binary classifier in BR-GroupNet can be computed as follows.

$$CL = loss + \sum\nolimits_{i=1}^{2} \alpha_i loss_i \tag{4}$$

$$CL1 = CE + \sum\nolimits_{i=1}^{2} \alpha_i CE_i \tag{5}$$

$$CL2 = FL + \sum\nolimits_{i=1}^{2} \alpha_i FL_i \tag{6}$$

Here α is a correlation coefficient value between every two labels, which is calculated by Pearson product-moment correlation coefficient (PMMC). In this work, we choose three chronic diseases as multi-label chronic disease prediction targets and only three independent binary classifiers are required. For the correlated loss of each independent classifier, the loss of each classifier itself as main loss, and the sum of product of two associated classifier losses and correlation coefficient values as the auxiliary loss. Hence, we set the range of parameter i from 1 to 2 in this work.

In this work, we use two different methods to calculate the correlated loss based on CE and FL, respectively, and named CL1 and CL2 as seen in Equations 5, 6. In order to validate whether selecting correlation coefficient value between two chronic diseases as hyper parameters of CL can work as we expected. The GroupNet architecture with correlated loss named BR-GroupNet-CL.

Optimization

In the training of CNN models, back-propagation method is carried out for the gradient. There are many hyper parameters of CNN models that need to be optimized. It is experimental, time-consuming and difficult to choose best hyper parameters. To initialize hyper parameters with less tuning in the training phase, Adam (Kingma and Ba, 2014; Chen et al., 2018; Reddi et al., 2018) optimizer is used for the gradient. It is a first-order gradient-based descent optimizer of stochastic objective function. Adam is based on adaptive estimates of lower-order moments and computes individual learning rates for different hyper parameter from estimates of first and second moments of the gradients. Comparing to stochastic gradient descent optimization (SGD) (Orr and Müller, 2003), Adam is more efficient, which requires less memory and training time.

The proper activation function also improves classification performance. There are several popular activation functions for neural networks, such as sigmoid, tanh, rectified linear unit (ReLU) (Nair and Hinton, 2010), Leaky ReLU (LeakyReLU) (Maas et al., 2013), Exponential Linear Units (ELU) (Clevert et al., 2015), Self-Normalizing Linear Units (SELU) (Klambauer et al., 2017), and so on. In this work, we test and compare all different activation functions in our datasets and choose the preferable one in all CNN models.

EXPERIMENTS AND EVALUATION

Experiment Setup

We implement all experiments based on the Scikit-learn library, WEKA software and Tensorflow platform. Scikit-learn library and WEKA are used to implement several machine learning methods, such as SVM, SMO, DT, Multilayer Perceptron (MLP). Tensorflow platform is used to implement deep learning methods, such as the proposed GroupNet architectures, IGCNet, GoogleNet (Szegedy et al., 2015), VGGNet (Simonyan and Zisserman, 2014), AlexNet, and deep neural network (DNN), Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU) (Shickel et al., 2018). The experiments run on a machine with Intel (R) 3.20 GHz CPU (i5-6500) and 8 GB RAM.

Furthermore, several experiments are conducted to select proper parameters based on the LP-GroupNet, such as batch size, learning rate, epochs, convolution kernel size, dropout rate, activation function, and focusing parameter γ in focal loss. In order to select preferable number of convolution units of group block for the GroupNet, we deploy three GroupNet architectures based on three different group blocks. The detail parameter setting of three different group blocks are {2, 2, 2, 1}, {2, 3, 3, 1} and {2, 4, 4, 1}, and **Figure 5** gives concrete CNN architectures of the three different GroupNet architectures, namely LP-GroupNet (**Figure 5A**), LP-GroupNet-3 (**Figure 5B**), and LP-GroupNet-4 (**Figure 5C**).

Evaluation Measures

Since multi-label classification can be converted into single-label multi-class classification and so the measures to evaluate single-label multi-class classification also can be used for this work. We adopt four common evaluation measures: F-score, accuracy, recall and precision measures to compare the performance of different methods for multi-label chronic disease classification. The accuracy is a measure to ensure that ratio of the prediction of true labels is correct. Precision is a measure system that is related to reproducibility, or how many predictions are correct. Recall is the fraction of true labels that were predicted correctly. F-score (F1) measure is the harmonic mean of precision and recall, and is a popular evaluation measure in the research area of data mining. Because the label distribution of chronic disease is skewed as described in Dataset and Data Preprocessing, weighted recall, weighted precision, weighted F-score are used to evaluate the classification performance of different methods. F1 evaluates the overall performance of the method better than accuracy, precision and recall according to related works (Tsoumakas and Katakis, 2007; Zhang and Zhou, 2014). Recall is an important evaluation measure in clinical. Different to normal F-score, the value of weighted F-score is not between weighted precision and weighed recall, instead it is smaller than both weighted precision and weighed recall. The following equations show how to calculate these values. TP, TN, FP, and FN are true positive, true negative, false positive, and false negative, respectively.

$$Accuracy = \frac{\sum_{i=1}^{l} \frac{TP_i+TN_i}{TP_i+FP_i+TN_i+FN_i}}{l} \tag{7}$$

$$Precision_{weighted} = \sum_{i=1}^{l} k_i \frac{TP_i}{TP_i+FP_i} \tag{8}$$

$$Recall_{weighted} = \sum_{i=1}^{l} k_i \frac{TP_i}{TP_i+FN_i} \tag{9}$$

$$\begin{aligned} F1_{weighted} &= \sum_{i=1}^{l} k_i \frac{\frac{TP_i}{TP_i+FP_i} \cdot \frac{TP_i}{TP_i+FN_i}}{\frac{TP_i}{TP_i+FP_i} + \frac{TP_i}{TP_i+FN_i}} \\ &= \sum_{i=1}^{l} k_i \frac{2Precision_i Recall_i}{Precision_i + \mathrm{Re}\,call_i} \end{aligned} \tag{10}$$

Accuracy, $Precision_{weighted}$, $\mathrm{Re}\,call_{weighted}$, and$F1_{weighted}$can be computed by Equations (7–10). $k_i$denotes the single labels accounted for the proportion of all labels, $l$is equal to 8 and i ranges 1–8.

RESULTS AND DISCUSSION

Hyper Parameter Selection

In this section, we present results of hyper parameter selection in both **Figures 6**, **7**. **Figure 6A** shows how accuracy changes with epochs, and epochs are set 1, 5, 10, 15, 20, 25, 30, 40, 50, and 100, respectively in experiments. When epochs is above certain epochs like 20, the performance of the LP-GroupNet actually decreases drastically due to over-fitting. It is evident that the LP-GroupNet achieves the best performance when the epochs is 20 as you can see from **Figure 6A**.

Figure 6B shows the relationship between accuracy with learning rate, respectively. We set the learning rate 0.05, 0.03, 0.02, 0.01, 0.005, 0.003, 0.002, 0.001, 0.0005, 0.0003, 0.0002, and 0.0001, respectively in the experiments. It is clear that the LP-GroupNet obtains the best performance when the learning rate is 0.002 according to **Figure 6B**.

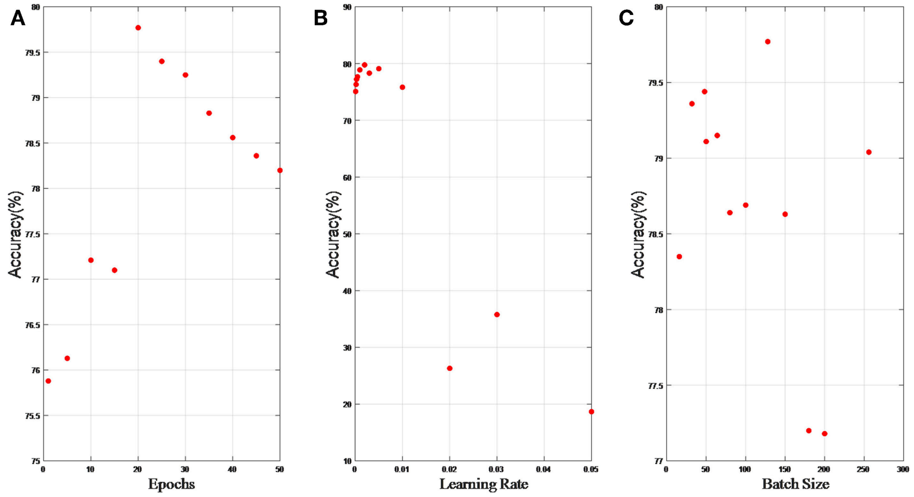


FIGURE 6 | (A) relationship between accuracy and epochs; **(B)** relationship between accuracy and learning rate; **(C)** relationship between accuracy and batch size.

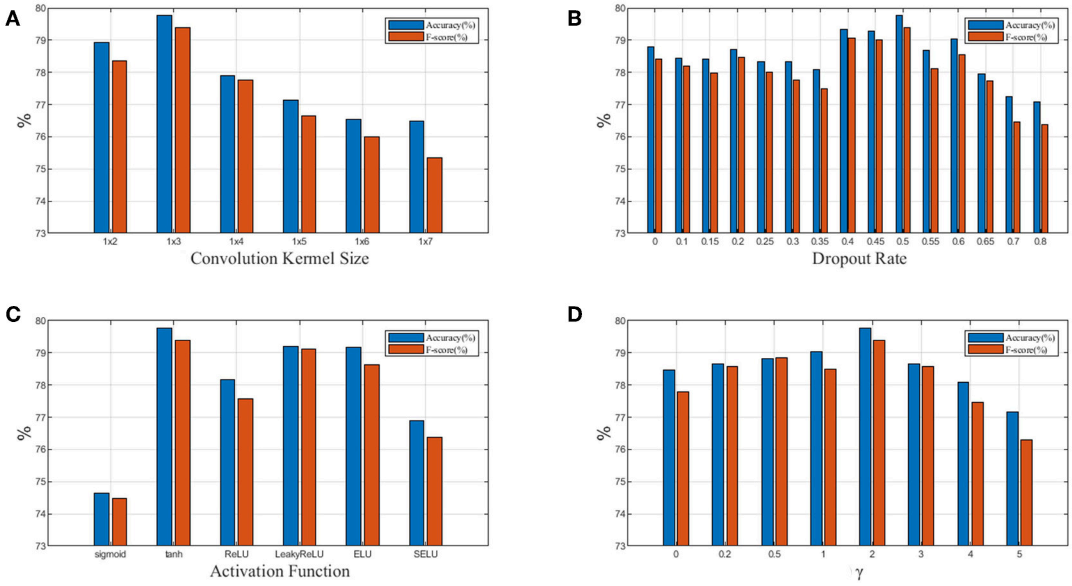


FIGURE 7 | (A) Relationship between performance and convolution kernel size; **(B)** Relationship between performance and dropout rate; **(C)** Relationship between performance and activation function; **(D)** Relationship between performance and focusing parameter γ in focal loss. Blue denotes accuracy and red denotes$F1_{weighted}$.

Figure 6C shows how batch size affects the LP-GroupNet performance, and we set batch size to 16, 32, 48, 50, 64, 80, 100, 128, 150, 180, 200, and 256, respectively. Accuracy changes with batch size quite significantly as you can see from **Figure 6C**. Results from the experiments show that the GroupNet achieves the best performance when batch size reaches 128.

In **Figure 7A**, we test 6 different convolution kernel sizes. The LP-GroupNet achieves best performance when convolution kernel size is 1 × 3. Furthermore, we also conclude that smaller convolution kernel works better than larger convolution kernel in previous works. **Figure 7B** presents how dropout rates influence the classification performance. It is shown that the LP-GroupNet gets the better performance when dropout rate is 0.5. It is difficult to find considerate dropout rate in the experiments as you can see from **Figure 7B**, because there is not a good way to find the best dropout rate theoretically except by experiments.

Figure 7C shows a performance comparison among six different activation functions: tanh, sigmoid, ReLU, LeakyReLU, ELU, and SELU. The tanh receives the best performance with 79.77% based on the GroupNet, while sigmoid receives the worst performance with 74.65%. It is noticeable that LeakyReLU

TABLE 1 | Comparison of Adam and SGD.

Optimizer	Accuracy (%)	$Precision_{weighted}$ (%)	$Recall_{weighted}$ (%)	$F1_{weighted}$ (%)
SGD	75.09	74.50	75.09	74.50
Adam	79.77	79.84	79.77	79.40

TABLE 2 | Comparison of different number of partition convolution units in group block.

Model	Accuracy (%)	$Precision_{weighted}$ (%)	$Recall_{weighted}$ (%)	$F1_{weighted}$ (%)
LP-GroupNet	79.77	79.84	79.77	79.40
GroupNet-3	79.66	79.42	79.66	79.22
GroupNet-4	79.20	78.88	78.20	78.88

TABLE 3 | Hyper-parameter settings of the GroupNet.

Hyper-parameter	Setting
Learning rate	0.002
Epochs	20
Batch size	128
Convolution kernel size	1 × 3
Dropout rate	0.5
Activation Function	tanh
γ	2
Optimizer	Adam
The number of partition convolution units	2

and ELU both get accuracy over 79%. In order to achieve considerable performance, the tanh function is more adaptive as activation function than others in this work. **Figure 7D** shows how focusing parameter γ in focal loss affects the LP-GroupNet performance and γ is set 0, 0.2, 0.5, 1.0, 2.0, 3.0, 4.0, and 5.0, respectively. When focusing parameter is 0, focal loss is equivalent to standard cross entropy loss. It is clear that it results in the best performance with the accuracy of 79.77% when focusing parameter γ is 2.

Table 1 gives a comparison between Adam optimizer and SGD optimizer. It is apparent that Adam optimizer outperforms SGD optimizer. Furthermore, SGD optimizer requires 160 epochs to achieve the accuracy at 75.09%, while Adam optimizer uses 20 epochs to achieve the accuracy 79.77%. With trading-off on training time and accuracy, Adam is selected as optimizer.

Table 2 presents the results for LP-GroupNet, LP-GroupNet-3, and LP-GroupNet-4. The results illuminate that the LP-GroupNet gets better performance than LP-GroupNet-3 and LP-GroupNet-4 models. It confirms that when the number of partition convolution units is 2 in group block, the GroupNet is able to handle the data more effectively and achieves the performance as we expected.

Table 3 lists the final optimal hyper-parameter settings.

TABLE 4 | Comparison of CNN models based on LP method.

Model	Accuracy (%)	$Precision_{weighted}$ (%)	$Recall_{weighted}$ (%)	$F1_{weighted}$ (%)
GroupNet	79.77	79.84	79.77	79.40
IGCNet	78.08	77.64	78.08	77.65
GoogleNet	78.56	79.02	78.56	78.41
AlexNet	76.28	77.03	76.28	76.10
VGGNet	78.17	77.79	78.17	77.46

TABLE 5 | Comparison of LP-GroupNet and BR-GroupNet.

Model	Accuracy (%)	$Precision_{weighted}$ (%)	$Recall_{weighted}$ (%)	$F1_{weighted}$ (%)
LP-GroupNet	79.77	79.84	79.77	79.40
BR-GroupNet	80.54	80.70	80.54	80.35

TABLE 6 | Comparison of different loss functions based on the BR-GroupNet.

Loss	Accuracy (%)	$Precision_{weighted}$ (%)	$Recall_{weighted}$ (%)	$F1_{weighted}$ (%)
CE	79.05	78.77	79.05	78.54
FL	80.54	80.70	80.54	80.35
CL1	79.66	80.59	79.66	79.30
CL2	81.13	81.37	81.13	81.02

TABLE 7 | Comparison of GroupNet model and other comparative methods.

Model	Accuracy (%)	$Precision_{weighted}$ (%)	$Recall_{weighted}$ (%)	$F1_{weighted}$ (%)
BR-GroupNet-CL	81.13	81.37	81.13	81.02
IGCNet	78.08	77.64	78.08	77.65
GoogleNet	78.56	79.02	78.56	78.41
AlexNet	76.28	77.03	76.28	76.10
VGGNet	78.17	77.79	78.17	77.46
DNN	71.10	75.70	71.12	72.61
LSTM	75.83	75.31	75.83	75.24
GRU	76.35	76.34	76.35	75.58
DT	77.26	77.12	77.34	77.12
MLP	74.94	74.40	74.95	74.40
SVM	48.89	42.2	49.91	41.6
SMO	70.12	67.60	70.12	67.42
ML-KNN	51.03	60.21	53.02	50.47
BPMLL	76.65	76.72	76.65	76.32

Comparison of Different Methods

Table 4 presents comparison results of the GroupNet and other CNN models based on LP method. The GroupNet achieves the best performance and increases 1.21% at least than other four CNN models on all evaluation measures.

It is observed that the BR-GroupNet model provides the accuracy with 80.54% in **Table 5**. It increases over 0.77% than LP-GroupNet on all evaluation measures and $F1_{weighted}$ receives the best improvement with 0.95%, which demonstrates that BR-GroupNet model is more suitable for this work than LP-GroupNet.

Table 6 presents a comparison among correlated loss and other loss functions based on the BR-GroupNet architecture. For convenience, cross entropy loss is named as CE in short, focal loss as FL, correlated loss based on cross entropy loss as CL1, and correlated loss based on focal loss as CL2.

It is obvious that CL2 gets the best accuracy with 81.13%. The results also demonstrate that CL works better than FL and CE based on the BR-GroupNet in this work, which increases approximately 0.6% on all metrics. The results from CL1 and CL2 demonstrate that correlation coefficient value between two chronic diseases is selected as hyper parameter of CL can work as we expected. Furthermore, FL achieves better performance than CE, which confirms that FL can improve classification performance by reducing the proportion of correctly classified instance loss in all loss in the training phase.

Table 7 presents the results for the BR-GroupNet-CL, four state-of-art CNN architectures, two RNN architectures (LSTM and GRU) and seven classical machine learning methods. According to these results, deep learning methods get better performance than classic machine learning methods generally, which show deep learning methods have great potentials in disease prediction. It is apparent that the BR-GroupNet-CL architecture provides the best performance among all of them on all metrics, while the SVM receives the worst performance. IGCNet, GoogleNet, AlexNet, VGGNet, LSTM, GRU, and BPMLL show similar performance and they all receive over 75% on all evaluation measures. According to the **Table 7**, BR-GroupNet-CL gets the best accuracy and $F1_{weighted}$ with 81.13 and 81.02%, respectively, and it increases 2.61% than other comparative methods which confirms that the proposed BR-GroupNet-CL is more able to receive considerable performance for multi-label chronic disease classification. Particularly, BR-GroupNet-CL model achieves $Re\,call_{weighted}$ with 81.13% and increases at least 2.57% comparing to other methods, which is a considerable improvement for disease classification clinically.

CONCLUSIONS

We propose a novel group block inspired by group convolution strategy to reduce the number of convolution parameters and improve the classification performance. Furthermore, we develop the GroupNet based on group block, then combine GroupNet with BR and LP methods for multi-label classification of chronic diseases, respectively. We present a cost sensitive learning algorithm named correlated loss to improve the performance. The results indicate that the proposed GroupNet gets the best accuracy with 81.13%, which is nearly 2.6% higher than all other comparison methods.

In the future work, we will focus on enhancing the learning ability of the CNN model and reduce over-fitting in the training. The transfer learning and adversarial learning methods will be applied to the model.

AUTHOR CONTRIBUTIONS

RL conceived the project. XZ implemented the algorithm and performed the computational analysis. SZ and HZ supervised the experiments. XZ, RL, SZ, and HZ drafted the manuscript. All authors revised and approved the final version of the manuscript.

FUNDING

The project is supported by the Program of Scientific and Technological Research of Henan Province, China, under Contract 152102210249, 192102310215, and Program of Key Science and Technological Research of Henan Province (18A520049).

ACKNOWLEDGMENTS

The authors thank the Collaborative Innovation Center on Internet Healthcare and Health Service of Henan Province, Zhengzhou University, for providing the medical records for analysis. The physical examination dataset used in this study is located at http://pinfish.cs.usm.edu/dnn/. There are two versions of the data available for download: a simple text file and an ARFF file for use with WEKA. Details about the format of the data are located on the webpage. The personally identifiable information from this dataset has been removed to ensure patient anonymity.

REFERENCES

Bouthillier, X., Konda, K., Vincent, P., and Memisevic, R. (2015). Dropout as data augmentation. *arXiv:1506.08700.*

Chen, X., Liu, S., Sun, R., and Hong, M. (2018). On the convergence of a class of adam-type algorithms for non-convex optimization. *arXiv:1808.02941.*

Clevert, D.-A., Unterthiner, T., and Hochreiter, S. (2015). Fast and accurate deep network learning by exponential linear units (ELUs). *CoRR abs/1511.07289.* Available online at: http://arxiv.org/abs/1511.07289

Gu, B., Sheng, V. S., and Li, S. (2015). "Bi-parameter space partition for cost-sensitive SVM," in *Proceedings of the 24th International Conference on Artificial Intelligence IJCAI'15* (Buenos Aires: AAAI Press), 3532–3539.

Hong, H., Liu, J., Bui, D. T., Pradhan, B., Acharya, T. D., Pham, B. T., et al. (2018). Landslide susceptibility mapping using J48 decision tree with adaboost, bagging and rotation forest ensembles in the guangchang area (China). *Catena* 163, 399–413. doi: 10.1016/j.catena.2018.01.005

Huang, G., Liu, Z., and Weinberger, K. Q. (2016). Densely connected convolutional networks. *CoRR abs/1608.06993*. Available online at: http://arxiv.org/abs/1608.06993

Iandola, F. N., Moskewicz, M. W., Ashraf, K., Han, S., Dally, W. J., and Keutzer, K. (2016). SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and < 0.5mb model size. *CoRR abs/1602.07360*. Available online at: http://arxiv.org/abs/1602.07360

Jin, M., Bahadori, M. T., Colak, A., Bhatia, P., Celikkaya, B., Bhakta, R., et al. (2018). Improving hospital mortality prediction with medical named entities and multimodal learning. *arXiv:1811.12276.*

Khan, S. H., Hayat, M., Bennamoun, M., Sohel, F. A., and Togneri, R. (2018). Cost-sensitive learning of deep feature representations from imbalanced data. *IEEE Trans Neural Netw Learn Syst.* 29, 3573–3587. doi: 10.1109/TNNLS.2017.2732482

Kingma, D. P., and Ba, J. (2014). Adam: a method for stochastic optimization. *arXiv:1412.6980*.

Klambauer, G., Unterthiner, T., Mayr, A., and Hochreiter, S. (2017). Self-normalizing neural networks. *CoRR abs/1706.02515*. Available online at: http://arxiv.org/abs/1706.02515

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems* 25, 1097–1105. doi: 10.1145/3065386

Lehnert, T., Heider, D., Leicht, H., Heinrich, S., Corrieri, S., Luppa, M., et al. (2011). Health care utilization and costs of elderly persons with multiple chronic conditions. *Med. Care Res. Rev.* 68, 387–420. doi: 10.1177/1077558711399580

Li, R., Liu, W., Lin, Y., Zhao, H., and Zhang, C. (2017a). An ensemble multilabel classification for disease risk prediction. *J Health Eng.* 2017:8051673. doi: 10.1155/2017/8051673

Li, R., Zhao, H., Lin, Y., Maxwell, A., and Zhang, C. (2017b). "Multi-label classification for intelligent health risk prediction," in *IEEE International Conference on Bioinformatics and Biomedicine* (Shenzhen: IEEE), 986–993.

Liang, Z., Zhang, G., Huang, J. X., and Hu, Q. V. (2014). "Deep learning for healthcare decision making with EMRs," in *Bioinformatics and Biomedicine (BIBM), 2014 IEEE International Conference on* (IEEE), 556–559. doi: 10.1109/BIBM.2014.6999219

Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2017). Focal loss for dense object detection. *arXiv:1708.02002*.

Lipton, Z. C., Kale, D. C., Elkan, C., and Wetzel, R. (2015). Learning to diagnose with LSTM recurrent neural networks. *arXiv:1511.03677.*

Ma, N., Zhang, X., Zheng, H.-T., and Sun, J. (2018). Shufflenet v2: practical guidelines for efficient cnn architecture design. *arXiv:1807.111645*.

Maas, A. L., Hannun, A. Y., and Ng, A. Y. (2013). "Rectifier nonlinearities improve neural network acoustic models," in *Proceedings of the International Conference on Machine* (Atlanta, GA), 3.

Maxwell, A., Li, R., Yang, B., Weng, H., Ou, A., Hong, H., et al. (2017). Deep learning architectures for multi-label classification of intelligent health risk prediction. *BMC Bioinform.* 18:523. doi: 10.1186/s12859-017-1898-z

Miotto, R., Li, L., Kidd, B. A., and Dudley, J. T. (2016). Deep patient: an unsupervised representation to predict the future of patients from the electronic health records. *Sci. Rep.* 6:26094. doi: 10.1038/srep26094

Mohamad Asri, M. N., Hashim, N. H., Mat Desa, W. N. S., and Ismail, D. (2018). Pearson Product Moment Correlation (PPMC) and Principal Component Analysis (PCA) for objective comparison and source determination of unbranded black ballpoint pen inks. *Austr. J. Forensic Sci.* 50, 323–340. doi: 10.1080/00450618.2016.1236292

Murphy, K. P. (2018). *Machine Learning: A Probabilistic Perspective (Adaptive Computation and Machine Learning Series)*. London, UK: The MIT Press.

Nair, V., and Hinton, G. E. (2010). "Rectified Linear Units Improve Restricted Boltzmann Machines," in *Proceedings of the 27th International Conference on International Conference on Machine Learning ICML'10* (Haifa: Omnipress), 807–814.

Orr, G. B., and Müller, K.-R. (2003). *Neural Networks: Tricks of the Trade*. Springer.

Reddi, S. J., Kale, S., and Kumar, S. (2018). "On the convergence of Adam and Beyond," in *International Conference on Learning Representations* (Vancouver, BC: Vancouver Convention Center).

Sandler, M., Howard, A. G., Zhu, M., Zhmoginov, A., and Chen, L.-C. (2018). Inverted residuals and linear bottlenecks: mobile networks for classification, detection and segmentation. *CoRR abs/1801.04381*. Available online at: http://arxiv.org/abs/1801.04381

Shanthi, M., Stephen, D., and Bo, N. (2015). Organizational update: the world health organization global status report on noncommunicable diseases 2014; one more landmark step in the combat against stroke and vascular disease. *Stroke* 46:e121. doi: 10.1161/STROKEAHA.115.008097

Shickel, B., Tighe, P. J., Bihorac, A., and Rashidi, P. (2018). Deep EHR: a survey of recent advances in deep learning techniques for electronic health record (EHR) analysis. *IEEE J. Biomed. Health Inform.* 22, 1589–1604. doi: 10.1109/JBHI.2017.2767063

Simonyan, K., and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556.*

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* 15, 1929–1958. Available online at: http://jmlr.org/papers/v15/srivastava14a.html

Sun, K., Li, M., Liu, D., and Wang, J. (2018). IGCV3: interleaved low-rank group convolutions for efficient deep neural networks. *CoRR abs/1806.00178.* Available online at: http://arxiv.org/abs/1806.00178

Szegedy, C., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., et al. (2015). "Going deeper with convolutions," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* 1–9. doi: 10.1109/CVPR.2015.7298594

Tsoumakas, G., and Katakis, I. (2007). Multi-label classification: an overview. *Int. J. Data Warehousing Mining* 3, 1–13. doi: 10.4018/jdwm.2007070101

Tsoumakas, G., and Vlahavas, I. (2007). "Random k-labelsets: an ensemble method for multilabel classification," in *European Conference on Machine Learning* (Springer), 406–417.

Weber, J. H., and Immink, K. A. S. (2018). Maximum likelihood decoding for gaussian noise channels with gain or offset mismatch. *IEEE Commun. Lett.* 22, 1128–1131. doi: 10.1109/LCOMM.2018.2809749

Xie, G., Wang, J., Zhang, T., Lai, J., Hong, R., and Qi, G.-J. (2018). IGCV2: interleaved structured sparse convolutional neural networks. *CVPR abs/1804.06202.* Available online at: http://arxiv.org/abs/1804.06202

Zhang, M.-L., and Zhou, Z.-H. (2006). Multilabel neural networks with applications to functional genomics and text categorization. *IEEE Transac. Knowl. Data Eng.* 18, 1338–1351. doi: 10.1109/TKDE.2006.162

Zhang, M.-L., and Zhou, Z.-H. (2014). A review on multi-label learning algorithms. *IEEE Transac. Knowl. Data Eng.* 26, 1819–1837. doi: 10.1109/TKDE.2013.39

Zhang, T., Qi, G.-J., Xiao, B., and Wang, J. (2017). "Interleaved group convolutions," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 4383–4392. doi: 10.1109/ICCV.2017.469

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

ORIGINAL RESEARCH
published: 09 July 2019
doi: 10.3389/fgene.2019.00600

Five-Feature Model for Developing the Classifier for Synergistic vs. Antagonistic Drug Combinations Built by XGBoost

Xiangjun Ji[1,2], Weida Tong[3], Zhichao Liu[3] and Tieliu Shi[1,4]**

[1] The Center for Bioinformatics and Computational Biology, Shanghai Key Laboratory of Regulatory Biology, Institute of Biomedical Sciences–School of Life Sciences, East China Normal University, Shanghai, China, [2] Guangdong Provincial Key Laboratory of Proteomics, School of Basic Medical Sciences, Southern Medical University, Guangzhou, China, [3] National Center for Toxicological Research, United States Food and Drug Administration, Jefferson, AR, United States, [4] National Center for International Research of Biological Targeting Diagnosis and Therapy/Guangxi Key Laboratory of Biological Targeting Diagnosis and Therapy Research/Collaborative Innovation Center for Targeting Tumor Diagnosis and Therapy, Guangxi Medical University, Nanning, China

OPEN ACCESS

Edited by:
Nora L. Nock,
Case Western Reserve University,
United States

Reviewed by:
Mohamed Diwan M. AbdulHameed,
Biotechnology HPC Software Applications Institute (BHSAI),
United States
Meijian Guan,
Independent Researcher, Frederick,
United States
Runyu Jing,
Sichuan University, China

****Correspondence:***
Zhichao Liu
zhichao.liu@fda.hhs.gov
Tieliu Shi
tlshi@bio.ecnu.edu.cn

Specialty section:
This article was submitted to Toxicogenomics,
a section of the journal
Frontiers in Genetics

Received: *06 November 2018*
Accepted: *05 June 2019*
Published: *09 July 2019*

Citation:
Ji X, Tong W, Liu Z and Shi T (2019) Five-Feature Model for Developing the Classifier for Synergistic vs. Antagonistic Drug Combinations Built by XGBoost.
Front. Genet. 10:600.
doi: 10.3389/fgene.2019.00600

Combinatorial drug therapy can improve the therapeutic effect and reduce the corresponding adverse events. *In silico* strategies to classify synergistic vs. antagonistic drug pairs is more efficient than experimental strategies. However, most of the developed methods have been applied only to cancer therapies. In this study, we introduce a novel method, XGBoost, based on five features of drugs and biomolecular networks of their targets, to classify synergistic vs. antagonistic drug combinations from different drug categories. We found that XGBoost outperformed other classifiers in both stratified fivefold cross-validation (CV) and independent validation. For example, XGBoost achieved higher predictive accuracy than other models (0.86, 0.78, 0.78, and 0.83 for XGBoost, logistic regression, naïve Bayesian, and random forest, respectively) for an independent validation set. We also found that the five-feature XGBoost model is much more effective at predicting combinatorial therapies that have synergistic effects than those with antagonistic effects. The five-feature XGBoost model was also validated on TCGA data with accuracy of 0.79 among the 61 tested drug pairs, which is comparable to that of DeepSynergy. Among the 14 main anatomical/pharmacological groups classified according to WHO Anatomic Therapeutic Class, for drugs belonging to five groups, their prediction accuracy was significantly increased (odds ratio < 1) or reduced (odds ratio > 1) (Fisher's exact test, $p < 0.05$). This study concludes that our five-feature XGBoost model has significant benefits for classifying synergistic vs. antagonistic drug combinations.

Keywords: drug combination, XGBoost classifier, synergistic drug pair, antagonistic drug pair, model performance

INTRODUCTION

The *de novo* drug discovery paradigm of "one drug, one target, and one disease" has been greatly challenged by the increasing rate of drug attrition in clinical trials and drug withdrawal due to severe adverse drug reactions (ADRs) at the post-marketing stage (Wood, 2006). Considering the complexity of disease etiology and pathogenesis, alternative drug development approaches such as drug combinations have been promoted to provide more effective and safer regimens (Flemming, 2014; Sarah, 2017). Combinatorial drug treatments could work synergistically to boost efficacy, or act additively or antagonistically to alleviate ADRs (Jia et al., 2009). Drug combinations have been widely used to counter drug resistance in cancer therapy (Webster, 2016). One example of this is the combination of docetaxel with two HER2 inhibitors (i.e., pertuzumab and trastuzumab) for treating HER2-positive metastatic breast cancer, which achieved an approximately 16-month improvement in overall survival (OS) compared with the conventional single treatment option (Swain et al., 2015). Synthetic lethality could be employed when discussing feasible therapeutic strategies for treating gastric cancer (Guo et al., 2017). Besides oncological drug development, the use of drug combinations is also a popular approach for antibacterial and antifungal therapy (Spitzer et al., 2011) and diabetes (Lu et al., 2017; Xu et al., 2017). For example, Hsp90 inhibitors and the antifungal drugs azoles were combined to treat patients infected with *Candida albicans* and *Saccharomyces cerevisiae* (Hill et al., 2013). As mentioned above, the use of drug combinations has also been applied to alleviate ADRs. One example is fixed-dose combination therapies for treating type 2 diabetes, which effectively eliminated the side effects of diabetes drugs such as cardiovascular toxicity and enhanced the efficacy (Bell, 2013).

Recent success in drug combinations has primarily been the result of serendipity or clinical observation, which is time-consuming and knowledge-driven (Foucquier and Guedj, 2015). Computational approaches offer a rational and exhaustive exploration of all possible drug combination opportunities by integrating different biomedical data profiles (Sun et al., 2013; Bulusu et al., 2016). Efforts have been made to develop *in silico* approaches to accelerate effective drug combination discovery. These computational approaches are mainly divided into three categories: transcriptomic profiles and cell-based drug sensitivity assay-based modeling, network/system biology-based approaches, and machine learning algorithms. For example, Preuer et al. (2018) developed a deep learning modeling named DeepSynergy to predict anti-cancer drug synergy by incorporating chemical and genomic data, yielding an AUC of 0.90. In addition, the predictive performance of DeepSynergy was also superior to that of other state-of-the-art methodologies, including random forest (RF), gradient boosting machine, support vector machine, and elastic net. The pros and cons of these *in silico* approaches have been intensively discussed elsewhere (Bulusu et al., 2016).

Questions have been raised about how to integrate the diversity of biological information into a framework to improve the performance of tools for predicting the efficacy of drug combinations. First, the current *in silico* drug combination models are mainly focused on the field of oncology (Sun et al., 2015; Preuer et al., 2018). There is thus a lack of *in silico* models to explore the opportunities for using drug combinations in other therapeutic categories such as pediatric and infectious diseases. Second, numerous accumulative biological datasets have been generated and become widely available, so a comprehensive assessment of the predictive power of diverse biological profiles is imperative to provide useful information for further model development. Finally, no approach at *in silico* modeling will provide universally valid results. Therefore, we need to carefully define the domain in which modeling results are applicable to maximize their utility. To address these unresolved issues, there is an urgent need for novel methodologies and model development strategies.

XGBoost as a machine learning algorithm has become well established in the machine learning community and gained a positive reputation through numerous machine learning challenges (Chen and Guestrin, 2016). XGBoost is an ensemble method based on gradient boosted trees. Considering the rationale behind XGBoost, it may be a promising algorithm to integrate diverse biological information seamlessly and yield satisfactory predictive results. To the best of our knowledge, the XGBoost methodology has not been applied to classify synergistic vs. antagonistic drug combinations.

In this research, the XGBoost methodology is intended to classify synergistic vs. antagonistic drug combinations. To investigate the potential for applying the XGBoost methodology, we employed five different data profiles, namely, chemical structure information, human phenotypic information, pathways, protein targets, and protein–protein interactions, for model development. The proposed XGBoost model was comprehensively assessed based on feature importance, performance metrics, and degree of overfitting. The model was also compared with state-of-the-art machine/deep learning algorithms including RF, logistic regression (LR), naïve Bayes (NB) classifier, and DeepSynergy. The domains to which the proposed XGBoost model is applicable were also investigated by ranking model performance across different therapeutic categories.

MATERIALS AND METHODS

The workflow of this study was illustrated in **Figure 1**, which included major four parts: data curation, feature extraction, model development, and model interpretation.

Data Curation

To curate the drug pairs with known combination effectiveness, three data resources including the Drug Combination Database (DCDB) (Liu et al., 2014), Therapeutic Target Database (TTD) (Zhu et al., 2010), and the literature in PubMed (Fiorini et al., 2017) were used.

The DCDB[1] is devoted to the research and development of multi-component drugs (Liu et al., 2014). The updated DCDB 2.0 collected 1,363 drug combinations (330 approved and 1,033 investigational, including 237 unsuccessful usages), involving 904 individual drugs and 805 targets. In this study, the combinatorial medical effectiveness of 655 drug combinations corresponding to 544 synergistic drug pairs and 111 antagonistic ones was retrieved from DCDB.

[1] http://www.cls.zju.edu.cn/dcdb/index.jsf (accessed April, 2019).

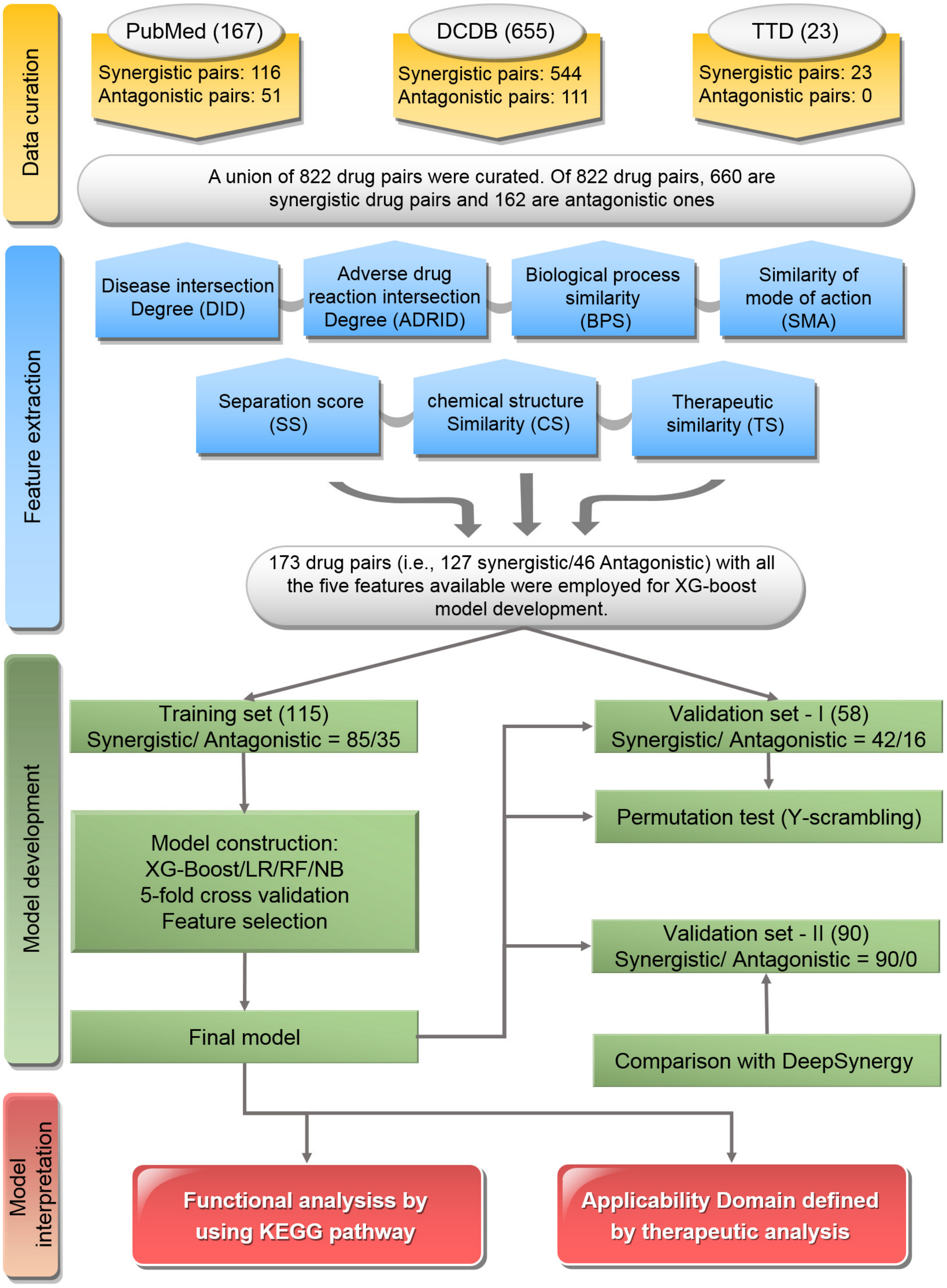

Data curation
PubMed (167)
Synergistic pairs: 116
Antagonistic pairs: 51
DCDB (655)
Synergistic pairs: 544
Antagonistic pairs: 111
TTD (23)
Synergistic pairs: 23
Antagonistic pairs: 0
A union of 822 drug pairs were curated. Of 822 drug pairs, 660 are synergistic drug pairs and 162 are antagonistic ones
Feature extraction
Disease intersection Degree (DID)
Adverse drug reaction intersection Degree (ADRID)
Biological process similarity (BPS)
Similarity of mode of action (SMA)
Separation score (SS)
chemical structure Similarity (CS)
Therapeutic similarity (TS)
173 drug pairs (i.e., 127 synergistic/46 Antagonistic) with all the five features available were employed for XG-boost model development.
Model development
Training set (115)
Synergistic/ Antagonistic = 85/35
Model construction:
XG-Boost/LR/RF/NB
5-fold cross validation
Feature selection
Final model
Validation set - I (58)
Synergistic/ Antagonistic = 42/16
Permutation test (Y-scrambling)
Validation set - II (90)
Synergistic/ Antagonistic = 90/0
Comparison with DeepSynergy
Model interpretation
Functional analysis by using KEGG pathway
Applicability Domain defined by therapeutic analysis

FIGURE 1 | Flowchart of the study: The workflow includes data curation, feature extraction, model development, and model interpretation.

Therapeutic Target Database[2] is a database to provide information about the known and explored therapeutic protein and nucleic acid targets, the targeted disease, pathway information, and the corresponding drugs directed at each of these targets. It contains 75 drug combinations. In this study, the combinatorial medical effectiveness of 23 drug combinations (e.g., 23 synergistic drug pairs vs. 0 antagonistic ones) were employed.

[2] http://bidd.nus.edu.sg/group/cjttd/TTD_HOME.asp

PubMed[3] comprises more than 28 million citations for the biomedical literature from MEDLINE, life science journals, and online books (Suarez-Almazor et al., 2000; Boddy, 2009). In this study, the combinatorial medical effectiveness of 167 drug combinations (e.g., 116 synergistic drug pairs vs. 51 antagonistic ones) was mined from PubMed with the Java library OpenNLP[4] for text mining (**Supplementary Table S1**).

Together, a union list of 822 drug pairs with known combinatorial medical effectiveness based on the three resources was obtained. Among them, 660 are synergistic drug pairs and 162 are antagonistic ones (**Supplementary Table S2**).

Feature Extraction

A list of seven features to describe the synergistic effect of drug pairs were generated in this study. These seven features were designed to comprehensively cover the molecular and phenotypic characteristics of drugs as well as their on/off targets. The details of these seven features are listed below:

(1) Disease intersection degree (DID): Drug–disease relationships were obtained from DrugBank (Wishart et al., 2018) and TTD (Li et al., 2018). DID represents the proportion of the same indications of two drugs. The higher the DID, the greater the proportion of the same indications of two drugs. The formula of DID is as follows:

$$DID_{\mathrm{a,b}} = \frac{D_{\mathrm{a}} \cap D_{\mathrm{b}}}{D_{\mathrm{a}} \cup D_{\mathrm{b}}} \tag{1}$$

Among these values, D_{a} and D_{b} represent the diseases treated by drugs *a* and *b*, respectively.

(2) Adverse drug reaction intersection degree (ADRID): ADRs were obtained from SIDER (Kuhn et al., 2016) and ADReCS (Cai et al., 2015). We defined ADRID as the Jaccard similarity between ADRs between two drugs. ADRID represents the proportion of the same ADRs of two drugs. The formula of ADRID is as follows:

$$ADRID_{\mathrm{a,b}} = \frac{ADR_{\mathrm{a}} \cap ADR_{\mathrm{b}}}{ADR_{\mathrm{a}} \cup ADR_{\mathrm{b}}} \tag{2}$$

Among them, ADR_{a} and ADR_{b} represent the ADRs of drugs *a* and *b*, respectively.

(3) Biological process similarity (BPS): BPS indicates the similarity between the biological processes for the interactants of two drugs. The higher the BPS, the greater the similarity of the biological process derived from the targets of two drugs. This feature was measured by GOSemSim (Yu et al., 2010). Targets, enzymes, and transporters of drugs were obtained from DrugBank (Wishart et al., 2018) and DGIDB (Cotto et al., 2018). BPS was calculated in R with the GOSemSim package which can be downloaded from http://www.bioconductor.org/packages/release/bioc/html/GOSemSim.html.

(4) Similarity of mode of action (SMA): This feature indicates the similarity of the mode (promotive/inhibitory) by which drugs act on the target in a drug pair. The higher the SMA, the greater the similarity of the mode (promotive/inhibitory) of action on the target of the two drugs. Drug–target interactions were obtained from DrugBank (Wishart et al., 2018) and DGIdb (Griffith et al., 2013). A protein interactive network with direction was obtained from KEGG (Kanehisa et al., 2016) and SIGNOR (Perfetto et al., 2016). All the interactions were directional and classified as promotive/inhibitory. The mode through which a chemical x acts on another non-adjacent chemical z depends on the relations of chemicals in all the shortest paths from x to z. If there are three chemicals, x, y, and z, with no direct link from x to z:

(a) If x promotes y and y promotes z, then x promotes z;
(b) If x promotes y and y inhibits z, then x inhibits z;
(c) If x inhibits y and y inhibits z, then x promotes z.

Then, the formula of SMA is as follows:

$$AMS_{\mathrm{a,b}} = \frac{\sum_{\mathrm{i}=1}^{M} \frac{\sum_{\mathrm{x}=1}^{X} c(a_{\mathrm{i}},b)_{\mathrm{x}}}{X} + \sum_{\mathrm{j}=1}^{N} \frac{\sum_{\mathrm{y}=1}^{Y} c(a,b_{\mathrm{j}})_{\mathrm{y}}}{Y}}{\sum_{\mathrm{i}=1}^{M} \frac{\left|\sum_{\mathrm{x}=1}^{X} c(a_{\mathrm{i}},b)_{\mathrm{x}}\right|}{X} + \sum_{\mathrm{j}=1}^{N} \frac{\left|\sum_{\mathrm{y}=1}^{Y} c(a,b_{\mathrm{j}})_{\mathrm{y}}\right|}{Y}} \tag{3}$$

a_{i} and b_{j} are the targets of drugs *a* and *b*, respectively. $c(a_{\mathrm{i}}, b)_{\mathrm{x}}$ is the coefficient of the shortest path x from a_{i} to b. The interpretation of $c(a_{\mathrm{i}}, b)$ also applies to $c(b_{\mathrm{j}}, a)$. If $c(a_{\mathrm{i}}, b)_{\mathrm{x}} = 1$, it means that the mode (promotive/inhibitory) of action of drug *b* on the target a_{i} through path *x* is the same as the mode (promotive/inhibitory) through which drug *a* acts on target a_{i}. If $c(a_{\mathrm{i}}, b)_{\mathrm{x}} = -1$, this means that the mode by which drug b acts on the target a_{i} through path *x* is the opposite of the mode by which drug *a* acts on target a_{i}. The numerator is normalized by the denominator in the formula. $SMA_{\mathrm{a,b}}$ ranges from -1 to 1. If the modes by which drug *b* acts on all the targets of drug *a* are the same as the modes by which drug *a* acts on them, $SMA_{\mathrm{a,b}} = 1$; alternatively, if the modes by which drug *b* acts on all the targets of drug *a* are the opposite of the modes by which drug *a* acts on them, $SMA_{\mathrm{a,b}} = -1$.

(5) Separation score (SS): This score is initially used to calculate module distances of two diseases, which is referred to as network separation (Menche et al., 2015). We first mapped all drug targets to the protein interaction network from InWeb_IM (Uhlik et al., 2016). In our model, separation score quantifies the network-based separation S_{ab} of two drugs a and b by comparing the mean shortest distances $<d_{\mathrm{aa}}>$ and $<d_{\mathrm{bb}}>$ between the respective drugs, to the mean shortest distance $<d_{\mathrm{ab}}>$ between their targets:

$$s_{\mathrm{ab}} = < d_{\mathrm{ab}} > - \frac{< d_{\mathrm{aa}} > + < d_{\mathrm{bb}} >}{2} \tag{4}$$

(6) Chemical structure similarity: The simplified molecular-input line-entry system (SMILES) is a specification in form of a line notation for describing the structure of chemical species using short ASCII strings (Weininger, 1988). SMILES information was obtained from DrugBank. Chemical structure similarity was calculated by Tanimoto similarity of SMILES in RDKit (Saubern et al., 2011).

(7) ATC similarity: We used the World Health Organization (WHO) ATC classification system (Skrbo et al., 2004). The ATC similarity between two drugs was induced from Gottlieb et al. (2012).

[3] https://www.ncbi.nlm.nih.gov/pubmed/

[4] http://opennlp.apache.org/

The calculated features were listed in **Supplementary Table S2**.

Model Development

The XGBoost Classifier

XGBoost (Extreme Gradient Boosting) is a machine learning technique for regression and classification problems based on the Gradient Boosting Decision Tree (GBDT) (Chen and Guestrin, 2016). The XGBoost model has been widely applied in all kinds of data mining fields for regression and classification, but has not yet been imported into the field of pharmacology. XGBoost is essentially an ensemble method based on gradient boosted tree (Friedman, 2001). In the regression tree, the inside nodes represent values for an attribute test and the leaf nodes with scores represent a decision. The result of the prediction is the sum of the scores predicted by K trees, as shown in the formula below:

$$\hat{y}_{\mathrm{i}} = \sum\nolimits_{\mathrm{k}=1}^{K} f_{\mathrm{k}}\left(x_{\mathrm{i}}\right), f_{\mathrm{k}} \in F \tag{5}$$

where x_{i} is the *i*-th training sample, $f_{\mathrm{k}}(x_{\mathrm{i}})$ is the score for the *k*-th tree, and F is the space of functions containing all regression trees. The objective function to be optimized is given by the following formula:

$$\mathrm{obj}\left(\theta\right) = \sum\nolimits_{\mathrm{i}=1}^{n} l\left(y_{\mathrm{i}}, \hat{y}_{\mathrm{i}}\right) + \sum\nolimits_{\mathrm{k}=1}^{K} \Omega\left(f_{\mathrm{k}}\right) \tag{6}$$

The former $\sum_{\mathrm{i}=1}^{n} l\left(y_{\mathrm{i}}, \hat{y}_{\mathrm{i}}\right)$ is a differentiable loss function that measures whether the model is suitable for training set data. The latter $\sum_{\mathrm{k}=1}^{K} \Omega\left(f_{\mathrm{k}}\right)$ is an item that punishes the complexity of the model. When the complexity of the model increases, the corresponding score is deducted.

In this study, variables input into the XGBoost classifier are the features of drug pairs and the variables that are output are the predicted classes and the corresponding possibilities of combinatorial medical effectiveness in a scale of 0∼1. The probability over 0.5 indicates that the combination is inclined to be synergistic, and the one under 0.5 means that the combination is inclined to be antagonistic. Some prediction values of drug combinations are around 0.5, which reflect that the combination is inclined to be additive.

Model Generation

(1) Division of training set and independent validation set: Of the 822 drug pairs curated with known combinatorial medical effectiveness, 173 drug pairs (synergistic drug pairs: antagonistic drug pairs ratio = 127:46) contain all the seven features described above were used for model construction and comparison since other models built by other classifiers (LR, NB, and RF) only accept the drug pairs with all features available as input.

Overall, 173 drug pairs were randomly divided into training set (approximately two-thirds, 115 drug pairs) and independent validation set-I (approximately one-third, 58 drug pairs) by keeping the original prevalence, which resulted in synergistic/antagonistic ratios of 85/30 and 42/16 in the training and validation sets, respectively (**Supplementary Tables S3**, **S4**).

To further verify the model performance of our developed model, we employed combination drugs used in TCGA project (The Cancer Genome Atlas Research Network et al., 2013). Specifically, we extracted the medical information of patients from The Cancer Genome Atlas (TCGA) project with the R package RTCGA[5]. Most of the patients were administered more than one drug, showing the necessity of multidrug therapy (**Supplementary Figure S1**). We consider that these patients had all undergone combinatorial therapy with synergistic effects. We screened out 659 patients who took just two kinds of drug with an overlap of at least 5 days, including 90 different drug combinations (**Supplementary Tables S3**, **S4**). The 90 drug combinations pairs were use as the independent validation-II.

(2) Feature selection: To compare the model performance with different combinations composed of seven preliminary features, XGBoost model were built with different feature combination, yielding 127 (i.e., $\sum_{\mathrm{i}=1}^{7} C_{7}^{i} = 127$) XGBoost models. The model performance of 127 XGBoost models were evaluated base on the average accuracy from 50 time of fivefold CV. The optimized feature combination was determined by the corresponding XGBoost model with highest accuracy, which was used as the final model for further analysis.

(3) Model evaluation: Six performance metrics were used including AUC, accuracy, sensitivity, specificity, negative predictive value (NPV), and positive predictive value (PPV) to evaluate the models. Synergistic combinations were classified as positive while antagonistic combinations were classified as negative. For training set, the average value of each performance metrics based on 50 runs of fivefold CV were presented. For independent validation set-I, six performance metrics were generated and further compared with the CV results, which was used to investigate whether the built model suffered over-fitness. To further investigate whether the XGBoost model performance was better than chance, a permutation test by using Y-scrambling strategy was implemented. Specifically, 2,000 permuted datasets were generated for the training set, in which the effect of drug pairs was randomly scrambled. For each permutation, the accuracy was calculated. Then, the *p*-value was calculated to assess the probability of the accuracy based on real data obtained by chance. For independent validation set-II, only the sensitivity was calculated since the comparison drug pairs are all synergistic.

(4) Comparison with state-of-the-art methods: To further compare the model performance of XGBoost with the state-of-the-art methods, four classifiers including RF, LR, NB classifier, and DeepSynergy (Preuer et al., 2018). The default parameters were used for LR, and NB with sklearn package in Python v3.5. For RF, we tested different numbers of estimators (trees) and features considered in each split. The performance is not well correlated with the hyperparameters. Thus, the performance of RF presented is generated based on default parameters. For DeepSynergy, 14 drug pairs are overlapped in the validation set-II and labeled with yellow background in **Supplementary Table S7**. DeepSynergy and our XGBoost were employed to compare their model performance with these drug pairs.

[5] https://rtcga.github.io/RTCGA/

Model Interpretation

Applicability Domain of the Developed XGBoost Model

Since the drug combination pairs curated cover a wide spectrum of different therapeutic categories, a defined applicability domain would be helpful for further application for various purpose. Therefore, those drug pairs with 50 correct or incorrect predictions were extracted based on the average accuracy of 50 runs of fivefold CV and further classified according to the second level of WHO Anatomic Therapeutic Class (ATC[6]) (Skrbo et al., 2004). Fisher's exact tests were performed on these drug pairs for each drug category. The odds ratio is calculated by dividing the ratio of a certain kind of drug in drug pairs with correct prediction to all drugs with correct prediction on the one hand by the ratio of a certain kind of drug in drug pairs with incorrect prediction to all drugs with incorrect prediction on the other.

Pathway Analysis

To determine the association between predictive accuracy and biological relevance of the drug targets, the targets belonging to those drug pairs with 50 correct or incorrect predictions stated above were extracted and mapped to pathways in KEGG for enrichment analysis, respectively (Kanehisa et al., 2016). The enrich pathways were adjusted p-values less than 0.01 were considered as statistically significant pathways.

Code Availability

The codes used for the generation of these features have been uploaded in https://github.com/514419407/Five-feature-Model-for-Predicting-the-Effects-of-Drug-Combinations-Built-by-XG Boost.git. XGBoost model was constructed by the xgboost package in Python. Other models built by other classifiers (LR, NB, and RF) were constructed by the sklearn package in Python. The xgboost and sklearn packages can be downloaded from https://pypi.org/. The values of all key hyperparameters of different algorithms are in **Supplementary Table S5**.

RESULTS

Feature Selection

Figure 2 shows the average accuracy from 50 repetitions of the fivefold CV for the feature selection process in the XGBoost models. A total of 127 (*i.e.*, $\sum_{i=1}^{7} C_7^i = 127$) XGBoost models were developed based on the different combination of the seven features. The performance of all XGBoost models roughly tend to be stable after the size of features combination reached five; further increasing the number of features did not change the model performance or slightly decreased the performance. Thus, the five features with the highest accuracy were selected for the construction of the XGBoost model. The optimized five features included DID, ADRID, BPS, SMA, and separation score.

To further investigate the performance contribution of each optimized features, the performance of the models constructed with different five feature combinations (one feature alone, leaving one feature out, and all five features) by the XGBoost classifier (**Table 1**). The results show that, among the metrics used for model evaluation, which include AUC, sensitivity, specificity, PPV, NPV, and accuracy, sensitivity achieved the best result in all models. The model with all the features showed the best performance, especially for specificity, which

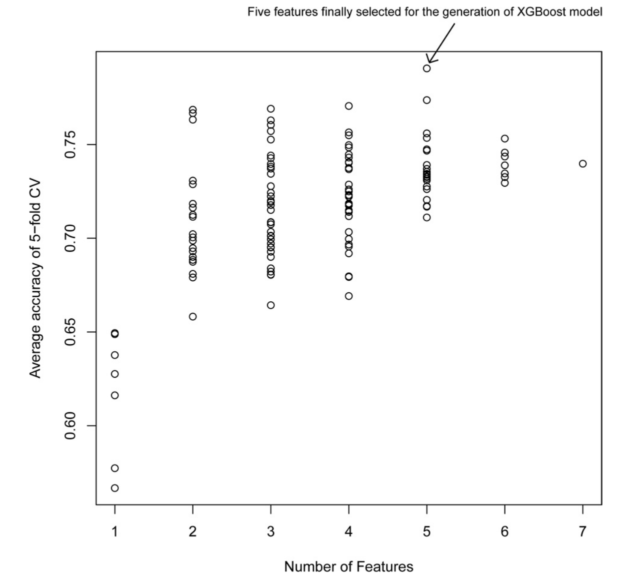


FIGURE 2 | Model performance of XG models with different feature combinations: The average accuracy of 50 runs of cross-validation (CV) was calculated for different XG models.

TABLE 1 | Performance of models constructed with different feature combinations (one feature alone, leave one feature out, and all features) by the XGBoost classifier.

Features	AUC	Sensitivity	Specificity	PPV	NPV	Accuracy
DID	0.46	0.79	0.03	0.70	0.53	0.65
ADRID	0.57	0.82	0.08	0.72	0.50	0.64
BPS	0.66	0.89	0.37	0.74	0.51	0.62
SMA	0.55	0.86	0.38	0.73	0.40	0.65
SS	0.60	0.87	0.30	0.75	0.48	0.56
No DID	0.74	0.89	0.46	0.73	0.62	0.70
No ADRID	0.71	0.90	0.30	0.73	0.55	0.69
No BPS	0.70	0.90	0.24	0.75	0.56	0.67
No SMA	0.73	0.92	0.40	0.74	0.58	0.68
No SS	0.73	0.91	0.43	0.73	0.59	0.68
All	0.77	0.95	0.63	0.82	0.67	0.79

DID, disease intersection degree; ADRID, adverse drug reaction intersection degree; BPS, biological process similarity; SMA, similarity of mode of action; SS, separation score. PPV, positive predictive value: TP/(TP+FP). NPV, negative predictive value: TN/(TN+FN). The average metrics of each model are displayed from 50 repetitions of the fivefold cross-validation (CV) carried out in the training set. The column names are the models made up of different combinations. The first five rows are models constructed with one feature alone; the middle five rows are models constructed when leaving one feature out; the last row is the model constructed with all five features.

[6] http://www.whocc.no/atcddd/

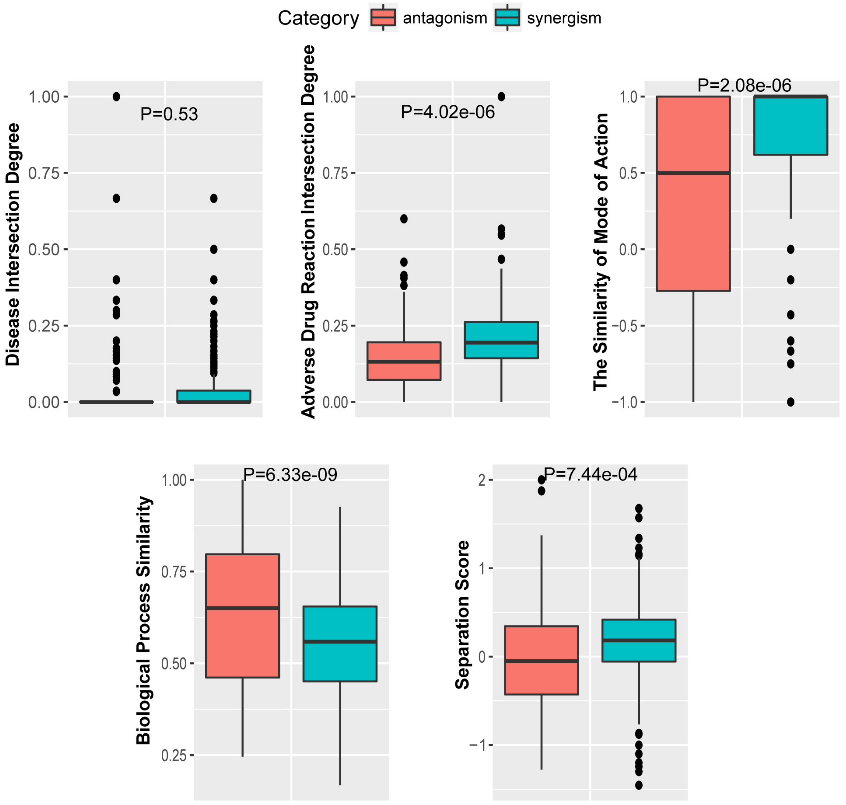


FIGURE 3 | The *t*-test for the five optimized features.

was much higher (at least 0.2) than those of the other models used in the comparison. Even for the SMA, the feature with the lowest *F*-score, the performance of all the leave-one-feature-out models was far behind that of the model built with all five features, showing the necessity of including all features in our model. The similar pattern was also observed based on Fisher's exact test. All these features were found to differ significantly between synergistic drug pairs and antagonistic drug pairs (*t*-test, $p < 0.05$), except for in the DID (*t*-test, $p = 0.53$) in the training set (**Figure 3** and **Supplementary Table S6**). Synergistic drug pairs show significantly higher ADRID, the SMA, and separation score, while showing significantly lower BPS (*t*-test, $p < 0.05$). The contribution of each feature to the XGBoost classifier is measured according to the intrinsic criterion of the XGBoost model, *F*-score (Chen and Guestrin, 2016) (**Figure 4**). The DID shows no significant difference between synergistic drug pairs and antagonistic drug pairs which is similar to its low contribution to the XGBoost classifier.

Model Performance for Validation Set-I

An extensive comparison of models built by XGBoost and other models was performed with all five features (see section "Materials and Methods"). **Figure 5** shows the six performance metrics based on 50 runs of in fivefold CV and independent validation (IV) for models built with different classifiers (**Supplementary Tables S6, S7**). The standard deviations of all CV metrics in the model built by XGBoost are all lower than those built by other classifiers when the values of all CV metrics in the XGBoost model are greater than those in models built by

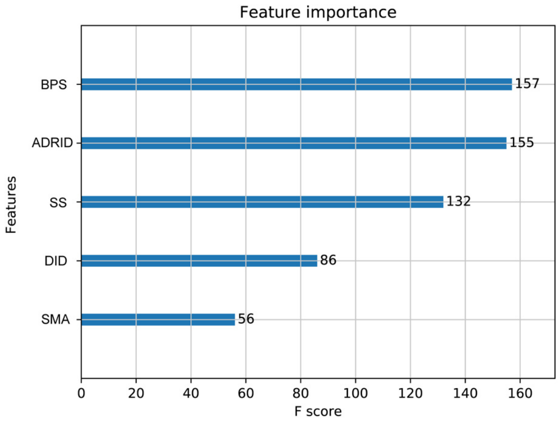


FIGURE 4 | Feature importance contributed to the XGBoost model measured by *F*-score: The average *F*-score of each model is displayed from 50 repetitions of the fivefold cross-validation (CV) carried out in the training set. Features in order of their contributions from large to small are as follows: BPS, biological process similarity; ADRID, adverse drug reaction intersection degree; SS, separation score; DID, disease intersection degree; SMA, the similarity of mode of action.

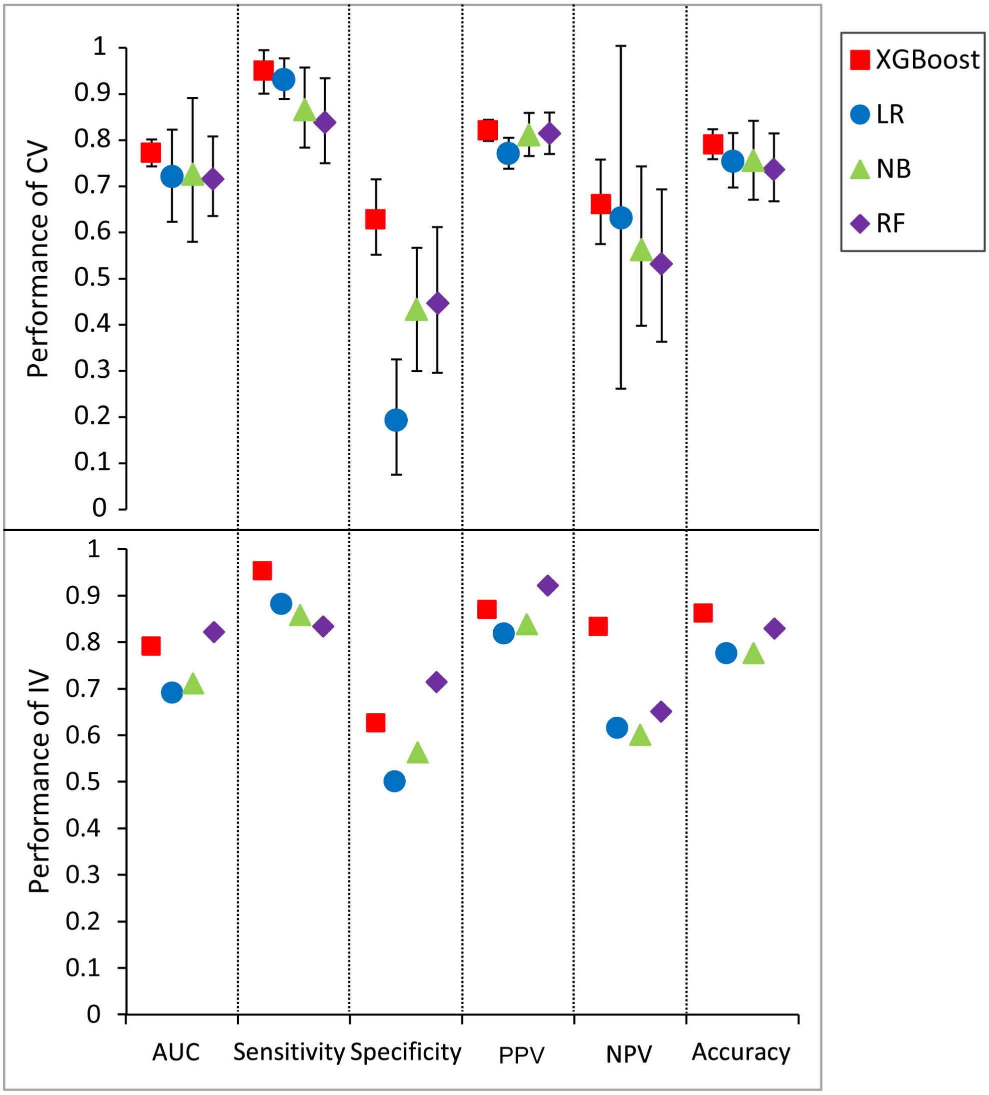


FIGURE 5 | Predictive values and standard deviation for six different metrics [AUC, sensitivity, specificity, positive predictive value (PPV), negative predictive value (NPV), accuracy] for the models constructed by different classifiers for both cross-validation (CV) and independent validation (IV). LR, logistic regression; NB, naïve Bayesian; RF, random forest.

other classifiers including RF, LR, and NB. A similar trend can also be observed for the other four IV performance metrics. For example, the values of four IV metrics in the XGBoost model are greater than those in models built by other classifiers. The values of accuracy in the XGBoost model in both CV and IV are at least 0.03 higher than those in models built by other classifiers. The performance ranks of the models on the IV set in terms of sensitivity and PPV are exactly consistent with the CV results. Since F1 score [2*((precision*recall)/(precision+recall))] conveys the balance between the precision and the recall, we also compared the values of F1 score among different models. The values of F1 score in the XGBoost model in both CV and IV are at least 0.025 higher than those in models built by other classifiers (**Supplementary Table S8**), with more true positives and fewer false negatives.

We also compared the difference in the six-performance metrics between the CV and IV (**Figure 6**), denoted as $|CV - IV|$, for the models constructed using four classifiers. The $|CV - IV|$ value measures the concordance; that is, a large $|CV - IV|$ value indicates either overtraining in the training model ($CV > IV$) or an unreliable extrapolation ($IV > CV$), since the performance of the internal validation should not be significantly better than that of the external validation. In addition to the best overall performance in both CV and IV, the XGBoost model also has the smallest $|CV - IV|$ values of the metrics (AUC, sensitivity, and specificity) among the different models.

Figure 7 shows the results of the permutation tests to assess whether the models predict the validation set better than would be expected by chance alone (see section "Materials and Methods"). If the predictive performance of

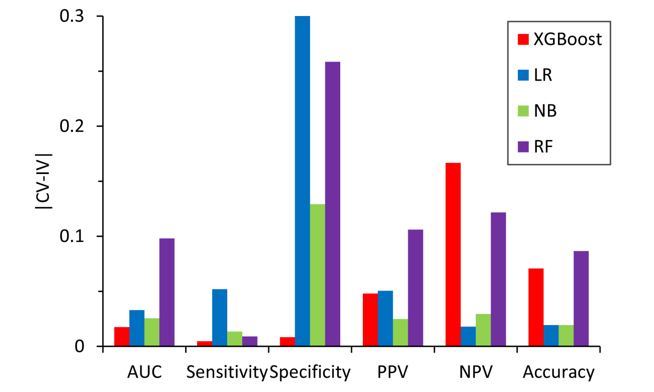


FIGURE 6 | Absolute difference, |CV – IV|, between the predictive performance for the six performance metrics in the fivefold cross-validation of the training set and the independent validation set for the models constructed by different classifiers.

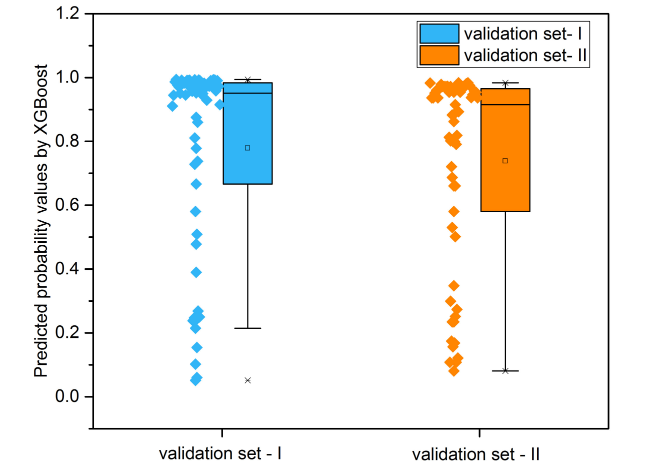


FIGURE 8 | The distribution of predicted probability values derived from XGBoost models for the two independent validation sets.

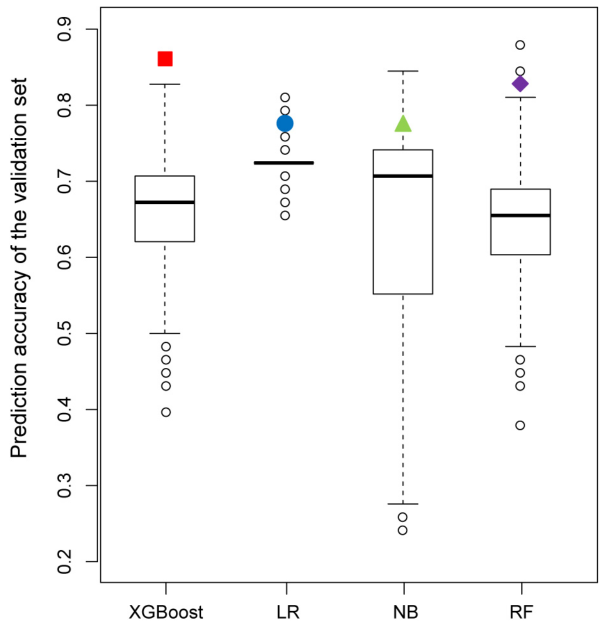


FIGURE 7 | Comparison of the prediction accuracy of the validation set between 2,000 models derived from a permutation test (randomly permuted) and the real dataset (colored solid dots).

a model measured by the real training set is not greater than that measured by the permutated training sets, we can conclude that the model measured by the real training set performs no better than the random results. Similar to the findings described in the previous section, the XGBoost model achieved the best performance in permutation tests. Unlike XGBoost, some of the values of prediction accuracy of the validation set derived from permutation tests were higher than those of the validation set derived from the real dataset in all other models.

Model Evaluation by Validation Set-II

To further confirm the performance of XGBoost, we tested the validation set obtained from TCGA with the XGBoost model. Of the 90 drug pairs involved in patients who underwent combinatorial therapy with a synergistic effect in TCGA (see section "Materials and Methods"), 61 drug pairs contained at least one feature in the XGBoost model. The XGBoost model classified these drug pairs with accuracy of 0.787 (**Supplementary Table S7**). These 61 drug pairs were used in 610 patients with 27 cancer types, with accuracy of over 0.94 calculated by the number of patients in TCGA, further demonstrating the robustness of our model.

To further validate the classification ability of the five-feature XGBoost model, we compared the prediction ability of the prediction ability between the five-feature XGBoost model and DeepSynergy. The original data profiles of the five-feature XGBoost model and DeepSynergy are different. To compare the prediction performance between the five-feature XGBoost model and DeepSynergy, we detected 14 overlapped drug pairs between the validation set-II of the five-feature XGBoost model and the prediction dataset of DeepSynergy since TCGA data are focused on cancer therapy. We displayed the predicted accuracy of the 14 overlapped drug pairs in 38 cell lines in DeepSynergy and in validation set-II. The highest accuracy could reach to 0.86 by using DeepSynergy, which is comparable to the accuracy (0.787) generated by XGBoost (**Supplementary Table S9**).

Distribution of Predicted Effectiveness by the Developed XGBoost Model

Figure 8 illustrated the distribution of possibility values for the two independent validation sets (**Supplementary Tables S6, S7**). The average possibility value of validation set-I and validation set-II since the drug pairs are 0.7788 ± 0.3074 and 0.7384 ± 0.3079. The large standard deviation indicated that the possibility values could be utilized to quantitatively reflect the effectiveness of drug combination pairs. Specifically, the scale of possibility is in a range of 0 to 1. The bigger possibility values indicated the higher synergistic effect. The lower possibility values mean the stronger antagonistic effect of drug pairs. The drug pairs with addictive effect were with possibility values around 0.5.

Applicability Domain of XGBoost Models

We then aimed to determine whether our model is able to classify drug pairs varied in different drug categories (see section "Materials and Methods"). Of the 822 drug pairs that we collected, the effectiveness of 745 drug pairs was correctly predicted at least once, while the effectiveness of 218 drug pairs was wrongly predicted at least once. The effectiveness of 604 drug pairs was correctly predicted in all 50 iterations, while the effectiveness of 77 drug pairs was wrongly predicted in all 50 iterations, showing the stability of the five-feature XGBoost model.

Drugs belonging to drug pairs with consistent prediction in all 50 iterations (both correct and incorrect predictions) were extracted to measure the predictive accuracy for different therapeutic categories. Among the 14-main anatomical/pharmacological groups classified based on WHO Anatomic Therapeutic Class (ATC, see text footnote 6), for drugs belonging to five groups, there are significant increases (odds ratio < 1) or reductions (odds ratio > 1) on their predictive accuracy (Fisher's exact test, $p < 0.05$) (**Table 2**, see section "Materials and Methods"). Specifically, among the drugs belonging to five groups, for antineoplastic and immunomodulating agents (abbreviated to L) and anti-infectives for systemic use (abbreviated to J), there is a significantly higher proportion of drugs in drug pairs with correctly predicted effectiveness than that of drugs in drug pairs with incorrectly predicted effectiveness (Fisher's exact test, $p < 0.01$; odds ratio < 1); for the drugs belonging to other three groups, there is a significantly lower proportion of drugs in drug pairs with correctly predicted effectiveness than that of drugs in drug pairs with incorrectly predicted effectiveness (Fisher's exact test, $p < 0.01$; odds ratio > 1).

TABLE 2 | Association of prediction accuracy and drug classification according to ATC codes by the stratified fivefold cross-validation.

Anatomical main group	Abbreviation	Odds ratio	*P*-value	#Drugs
Antineoplastic and immunomodulating agents	L	0.20	0.00	218
Nervous system	N	2.19	0.00	151
Various	V	4.43	0.00	27
Anti-infectives for systemic use	J	0.41	0.01	86
Alimentary tract and metabolism	A	1.84	0.03	50
Musculo-skeletal system	M	2.00	0.07	24
Respiratory system	R	1.77	0.10	32
Genito urinary system and sex hormones	G	1.61	0.19	33
Blood and blood forming organs	B	0.27	0.24	28
Antiparasitic products, insecticides and repellents	P	1.68	0.27	21
Dermatologicals	D	1.16	0.60	48
Sensory organs	S	1.09	0.76	60
Cardiovascular system	C	1.04	0.89	99
Systemic hormonal preparations, excl. sex hormones and insulins	H	0.86	1.00	8

The table is sorted according to P-values from low to high. The employed drugs belong to drug pairs with consistent prediction in all 50 iterations (both correct and incorrect predictions).

Associating Pathways With the Potential of the Five-Feature XGBoost Model

We next investigated whether our model can classify synergistic vs. antagonistic drug pairs with targets belonging to different pathways (see section "Materials and Methods"). We enriched the targets of drugs in correctly and incorrectly predicted drug pairs to 139 and 96 KEGG pathways (Bonferroni, *p*-value < 0.01), respectively (Kanehisa et al., 2016). Forty-three pathways exclusively belonged to the correctly predicted drug pairs (**Table 3**). The results of pathway analysis correspond to the results of drug category analysis. A number of pathways are associated with antineoplastic and immunomodulating agents, anti-infectives for systemic use including for malaria (Nosten and White, 2007), and bacterial invasion of epithelial cells.

DISCUSSION

The five-feature XGBoost model is an important advance for the classification of synergistic and antagonistic drug pairs. Classifying synergistic vs. antagonistic drug pairs experimentally is time-consuming and labor-intensive. *In silico* methods can thus be of tremendous benefit in this field of study. In this paper, we propose a model for efficiently classifying synergistic and antagonistic drug pairs. Its comparison with other models showed that it confers major advantages in accurately classifying synergistic vs. antagonistic drug pairs in combination, both with and without the existence of all five features.

With the extremely low $|\text{CV} - \text{IV}|$ value of sensitivity and the highest values in sensitivity and accuracy received from the XGBoost classifier, the five-feature XGBoost model shows much greater ability to predict the effects of combinatorial therapies with synergistic effects than those with antagonistic effects. Thus, our model is reliable for use as a filter to generate candidates of synergistic drug pairs. For example, the combination of caffeine and hexobarbital is an antagonistic drug pair that was wrongly classified as a synergistic drug pair by our model. This may have been due to the lack of feature values (DID and ADRID) in this drug pair.

According to our research, our model is preferable to classify synergistic vs. antagonistic drug pairs composed of antineoplastic and immunomodulating agents, anti-infectives for systemic use

TABLE 3 | Forty-three pathways exclusively belonging to correctly predicted drug pairs.

Pathway name	#Gene	*p*-Value
Proteasome	40	3.06E-54
Cytokine–cytokine receptor interaction	59	3.10E-31
Jak-STAT signaling pathway	35	2.27E-18
Epithelial cell signaling in *Helicobacter pylori* infection	24	2.72E-17
Leukocyte transendothelial migration	29	2.25E-16
NOD-like receptor signaling pathway	21	2.76E-15
Arrhythmogenic right ventricular cardiomyopathy (ARVC)	22	5.84E-14
Shigellosis	20	1.53E-13
Hematopoietic cell lineage	21	3.44E-11
African trypanosomiasis	14	1.37E-10
Malaria	16	2.57E-10
Rheumatoid arthritis	20	6.71E-10
Adherens junction	18	9.96E-10
Base excision repair	13	1.15E-09
PPAR signaling pathway	16	5.14E-08
Dorso-ventral axis formation	10	1.70E-07
Bacterial invasion of epithelial cells	15	4.84E-07
RIG-I-like receptor signaling pathway	15	5.97E-07
Wnt signaling pathway	21	1.29E-06
Protein digestion and absorption	15	3.99E-06
Arginine and proline metabolism	12	1.23E-05
Axon guidance	18	1.60E-05
Parkinson's disease	17	9.34E-05
Caffeine metabolism	5	9.90E-05
One carbon pool by folate	7	0.0001
Nucleotide excision repair	10	0.0001
Taste transduction	10	0.0006
Vibrio cholerae infection	10	0.0009
Tyrosine metabolism	8	0.0054
Type I diabetes mellitus	8	0.0078
Protein processing in endoplasmic reticulum	16	0.0094
ECM–receptor interaction	11	0.0105
Terpenoid backbone biosynthesis	5	0.0115
Nicotinate and nicotinamide metabolism	6	0.0127
Vitamin digestion and absorption	6	0.0127
Fat digestion and absorption	8	0.0129
DNA replication	7	0.0176
Allograft rejection	7	0.0189
Renin–angiotensin system	5	0.0189
Graft-versus-host disease	7	0.0378
Autoimmune thyroid disease	8	0.0378
Pyruvate metabolism	7	0.0378
Glycerophospholipid metabolism	10	0.0378

(**Table 2**). This may be due to the fact that cancer patients receive combinatorial drug therapy with targeted drugs in some circumstances (Al-Lazikani et al., 2012). The results of pathway analysis correspond to the results of drug category analysis. For example, malaria is treated by anti-infectives for systemic use and a pathway in KEGG belonging to the correctly predicted drug pairs. The reason for the excellent performance of the five-feature XGBoost model in malaria is according to the performance in anti-infectives for systemic use (**Table 2**) and malaria pathway (**Table 3**) that our prediction model follows the rules of combinatorial therapy for malaria of reducing the risk of treatment failure and reducing the side effects (Nosten and White, 2007).

Besides the advantages stated above, XGBoost can be constructed and performs prediction when drug pairs do not contain all five features, so it is more practical than other models as, among our 822 collected known drug pairs, only 173 contain all five features (**Supplementary Table S2**).

The five-feature XGBoost model contains relatively few features compared with other models (Sun et al., 2015). However, the features in our model are ubiquitous among drugs and other molecules potentially available for medical usage with vital medical significance. Intriguingly, our synergistic drug pairs show no significant difference from antagonistic drug pairs according to DID. This may be because not all the indications of the drug have been detected yet. In addition, although the SMA uses more precise information (promotive/inhibitory drug–target and protein–protein relationships) than other features, it makes the smallest contribution to our model. This may be due to the fewer related data.

It is worthwhile to consider some additional studies to further our knowledge and improve the prediction results from this study. First, the current *in silico* drug combination models are mainly focused on the field of oncology. There is thus a lack of *in silico* models to explore the opportunities for using drug combinations in other therapeutic categories such as pediatric and infectious diseases. Second, numerous accumulative biological datasets have been generated and become widely available, so a comprehensive assessment of the predictive power of diverse biological profiles is imperative to provide useful information for further model development. Third, the fine-tuning hyperparameters of machine-learning algorithm such as RF may provide improved model performance, however, it is not the focus of current study. Final, some novel algorithms for drug combination effectiveness prediction such as TreeCombo is worth exploring for better prediction results (Janizek et al., 2018).

CONCLUSION

In conclusion, we applied one machine-learning methodology, XGBoost, to classify the effects of drug combinations, which was greatly successful. In future work, deep learning algorithm such as RNN is also worth investigating for potential performance improvement. Although some other important features such as gene expression are not incorporated into our model (Sun et al., 2015), it may make a major contribution to predicting the effects of drug combinations.

AUTHOR CONTRIBUTIONS

ZL and TS designed the study. ZL and XJ performed the data analysis and wrote the manuscript. TS, XJ, ZL, and WT revised the manuscript. All authors read and approved the final manuscript.

FUNDING

This work was supported by the National High Technology Research and Development Program of China (Grant Nos. 2015AA020108 and 2016YFC0902100), the China Human Proteome Project (Grant Nos. 2014DFB30010 and 2014DFB30030), the National Science Foundation of China (Grant Nos. 31671377, 31401133, 31771460, and 91629103), and the 111 Project (Grant No. B14019).

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fgene.2019.00600/full#supplementary-material

FIGURE S1 | The distribution of patient sample size with different numbers of drugs during medical therapy from TCGA.

TABLE S1 | Information on 167 drug combinations retrieved from PubMed.

TABLE S2 | Features and real effectiveness of 822 known drug pairs.

TABLE S3 | Patients who took just two kinds of drugs with an overlap of at least 5 days from TCGA.

TABLE S4 | Tumor types included in **Supplementary Table S2**.

TABLE S5 | Key hyperparameters used in different models.

TABLE S6 | Features and real effectiveness used in the training set and validation set-1.

TABLE S7 | Features, real effect, and predicted effect of 61 drug pairs from TCGA based on the five-feature XGBoost model.

TABLE S8 | The values of F1 score in CV and IV.

TABLE S9 | Accuracy of the 14 overlapped drug pairs in 38 cell lines in DeepSynergy and in validation set-2.

REFERENCES

Al-Lazikani, B., Banerji, U., and Workman, P. (2012). Combinatorial drug therapy for cancer in the post-genomic era. *Nat. Biotechnol.* 30, 679–692. doi: 10.1038/nbt.2284

Bell, D. S. H. (2013). Combine and conquer: advantages and disadvantages of fixed-dose combination therapy. *Diabetes Obes. Metab.* 15, 291–300. doi: 10.1111/dom.12015

Boddy, K. (2009). When is a search not a search? A comparison of searching the AMED complementary health database via EBSCOhost, OVID and DIALOG. *Health Info. Libr. J.* 26, 126–135. doi: 10.1111/j.1471-1842.2008.00785.x

Bulusu, K. C., Guha, R., Mason, D. J., Lewis, R. P. I., Muratov, E., Kalantar Motamedi, Y., et al. (2016). Modelling of compound combination effects and applications to efficacy and toxicity: state-of-the-art, challenges and perspectives. *Drug Discov. Today* 21, 225–238. doi: 10.1016/j.drudis.2015.09.003

Cai, M. C., Xu, Q., Pan, Y. J., Pan, W., Ji, N., Li, Y. B., et al. (2015). ADReCS: an ontology database for aiding standardization and hierarchical classification of adverse drug reaction terms. *Nucleic Acids Res.* 43, D907–D913. doi: 10.1093/nar/gku1066

Chen, T., and Guestrin, C. (2016). "XGBoost: a scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, (San Francisco, CA: ACM).

Cotto, K. C., Wagner, A. H., Feng, Y. Y., Kiwala, S., Coffman, A. C., Spies, G., et al. (2018). DGIdb 3.0: a redesign and expansion of the drug-gene interaction database. *Nucleic Acids Res.* 46, D1068–D1073. doi: 10.1093/nar/gkx1143

Fiorini, N., Lipman, D. J., and Lu, Z. (2017). Towards PubMed 2.0. *eLife* 6:e28801. doi: 10.7554/eLife.28801

Flemming, A. (2014). Finding the perfect combination. *Nat. Rev. Drug Discov.* 14:13. doi: 10.1038/nrd4524

Foucquier, J., and Guedj, M. (2015). Analysis of drug combinations: current methodological landscape. *Pharmacol. Res. Perspect.* 3:e00149. doi: 10.1002/prp2.149

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Ann. Statist.* 29, 1189–1232.

Gottlieb, A., Stein, G. Y., Oron, Y., Ruppin, E., and Sharan, R. (2012). INDI: a computational framework for inferring drug interactions and their associated recommendations. *Mol. Syst. Biol.* 8:592. doi: 10.1038/msb.2012.26

Griffith, M., Griffith, O. L., Coffman, A. C., Weible, J. V., Mcmichael, J. F., Spies, N. C., et al. (2013). DGIdb - Mining the druggable genome. *Nat. Methods* 10, 1209–1210. doi: 10.1038/nmeth.2689

Guo, J., Yu, W., Su, H., and Pang, X. (2017). Genomic landscape of gastric cancer: molecular classification and potential targets. *Sci. China Life Sci.* 60, 126–137. doi: 10.1007/s11427-016-0034-1

Hill, J. A., Ammar, R., Torti, D., Nislow, C., and Cowen, L. E. (2013). Genetic and genomic architecture of the evolution of resistance to antifungal drug combinations. *PLoS Genet.* 9:e1003390. doi: 10.1371/journal.pgen.1003390

Janizek, J. D., Celik, S., and Lee, S.-I. (2018). Explainable machine learning prediction of synergistic drug combinations for precision cancer medicine. *bioRxiv* [Preprint]. doi: 10.1101/331769

Jia, J., Zhu, F., Ma, X., Cao, Z. W., Li, Y. X., and Chen, Y. Z. (2009). Mechanisms of drug combinations: interaction and network perspectives. *Nat. Rev. Drug Discov.* 8, 111–128. doi: 10.1038/nrd2683

Kanehisa, M., Sato, Y., Kawashima, M., Furumichi, M., and Tanabe, M. (2016). KEGG as a reference resource for gene and protein annotation. *Nucleic Acids Res.* 44, D457–D462. doi: 10.1093/nar/gkv1070

Kuhn, M., Letunic, I., Jensen, L. J., and Bork, P. (2016). The SIDER database of drugs and side effects. *Nucleic Acids Res.* 44, D1075–D1079. doi: 10.1093/nar/gkv1075

Li, Y. H., Yu, C. Y., Li, X. X., Zhang, P., Tang, J., Yang, Q., et al. (2018). Therapeutic target database update 2018: enriched resource for facilitating bench-to-clinic research of targeted therapeutics. *Nucleic Acids Res.* 46, D1121–D1127. doi: 10.1093/nar/gkx1076

Liu, Y., Wei, Q., Yu, G., Gai, W., Li, Y., and Chen, X. (2014). DCDB 2.0: a major update of the drug combination database. *Database* 2014:bau124. doi: 10.1093/database/bau124

Lu, J., Xia, Q., and Zhou, Q. (2017). How to make insulin-producing pancreatic beta cells for diabetes treatment. *Sci. China Life Sci.* 60, 239–248. doi: 10.1007/s11427-016-0211-3

Menche, J., Sharma, A., Kitsak, M., Ghiassian, S. D., Vidal, M., Loscalzo, J., et al. (2015). Uncovering disease-disease relationships through the incomplete interactome. *Science* 347:1257601. doi: 10.1126/science.1257601

Nosten, F., and White, N. J. (2007). Artemisinin-based combination treatment of falciparum malaria. *Am. J. Trop. Med. Hyg.* 77, 181–192. doi: 10.4269/ajtmh.2007.77.181

Perfetto, L., Briganti, L., Calderone, A., Cerquone Perpetuini, A., Iannuccelli, M., Langone, F., et al. (2016). SIGNOR: a database of causal relationships between biological entities. *Nucleic Acids Res.* 44, D548–D554. doi: 10.1093/nar/gkv1048

Preuer, K., Lewis, R. P. I., Hochreiter, S., Bender, A., Bulusu, K. C., and Klambauer, G. (2018). DeepSynergy: predicting anti-cancer drug synergy with deep learning. *Bioinformatics* 34, 1538–1546. doi: 10.1093/bioinformatics/btx806

Sarah, C. (2017). Identifying synergistic drug combinations. *Nat. Rev. Drug Discov.* 16:314. doi: 10.1038/nrd.2017.76

Saubern, S., Guha, R., and Baell, J. J. (2011). KNIME workflow to assess PAINS filters in SMARTS format. Comparison of RDKit and indigo cheminformatics libraries. *Mol. Inform.* 30, 847–850. doi: 10.1002/minf.201100076

Skrbo, A., Begovic, B., and Skrbo, S. (2004). Classification of drugs using the ATC system (Anatomic, Therapeutic, Chemical Classification) and the latest changes. *Med. Arh.* 58(1 Suppl. 2), 138–141.

Spitzer, M., Griffiths, E., Blakely, K. M., Wildenhain, J., Ejim, L., Rossi, L., et al. (2011). Cross-species discovery of syncretic drug combinations that potentiate the antifungal fluconazole. *Mol. Syst. Biol.* 7:499. doi: 10.1038/msb.2011.31

Suarez-Almazor, M. E., Belseck, E., Homik, J., Dorgan, M., and Ramos-Remus, C. (2000). Identifying clinical trials in the medical literature with electronic databases: MEDLINE alone is not enough. *Control. Clin. Trials* 21, 476–487. doi: 10.1016/s0197-2456(00)00067-2

Sun, X., Vilar, S., and Tatonetti, N. P. (2013). High-throughput methods for combinatorial drug discovery. *Sci. Transl. Med.* 5:205rv201.

Sun, Y., Sheng, Z., Ma, C., Tang, K., Zhu, R., Wu, Z., et al. (2015). Combining genomic and network characteristics for extended capability in predicting synergistic drugs for cancer. *Nat. Commun.* 6:8481. doi: 10.1038/ncomms9481

Swain, S. M., Baselga, J., Kim, S.-B., Ro, J., Semiglazov, V., Campone, M., et al. (2015). Pertuzumab, trastuzumab, and docetaxel in HER2-positive metastatic breast cancer. *N. Engl. J. Med.* 372, 724–734.

The Cancer Genome Atlas Research Network, Chang, K., Creighton, C. J., Davis, C., Donehower, L., Drummond, J., et al. (2013). The cancer genome atlas pan-cancer analysis project. *Nat. Genet.* 45, 1113–1120. doi: 10.1038/ng.2764

Uhlik, F., Kosovan, P., Zhulina, E. B., and Borisov, O. V. (2016). Charge-controlled nano-structuring in partially collapsed star-shaped macromolecules. *Soft Matter* 12, 4846–4852. doi: 10.1039/c6sm00109b

Webster, R. M. (2016). Combination therapies in oncology. *Nat. Rev. Drug Discov.* 15, 81–82.

Weininger, D. (1988). SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *J. Chem. Inf. Comput. Sci.* 28, 31–36. doi: 10.1093/bioinformatics/btn181

Wishart, D. S., Feunang, Y. D., Guo, A. C., Lo, E. J., Marcu, A., Grant, J. R., et al. (2018). DrugBank 5.0: a major update to the DrugBank database for 2018. *Nucleic Acids Res.* 46, D1074–D1082. doi: 10.1093/nar/gkx1037

Wood, A. J. (2006). A proposal for radical changes in the drug-approval process. *N. Engl. J. Med.* 355, 618–623. doi: 10.1056/nejmsb055203

Xu, W., Mu, Y., Zhao, J., Zhu, D., Ji, Q., Zhou, Z., et al. (2017). Efficacy and safety of metformin and sitagliptin based triple antihyperglycemic therapy (STRATEGY): a multicenter, randomized, controlled, non-inferiority clinical trial. *Sci. China Life Sci.* 60, 225–238. doi: 10.1007/s11427-016-0409-7

Yu, G., Li, F., Qin, Y., Bo, X., Wu, Y., and Wang, S. (2010). GOSemSim: an R package for measuring semantic similarity among GO terms and gene products. *Bioinformatics* 26, 976–978. doi: 10.1093/bioinformatics/btq064

Zhu, F., Han, B., Kumar, P., Liu, X., Ma, X., Wei, X., et al. (2010). Update of TTD: therapeutic target database. *Nucleic Acids Res.* 38, D787–D791. doi: 10.1093/nar/gkp1014

Disclaimer: The views presented in this article do not necessarily reflect current or future opinion or policy of the United States Food and Drug Administration. Any mention of commercial products is for clarification and not intended as an endorsement.

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

frontiers in Physiology

ORIGINAL RESEARCH
published: 13 August 2019
doi: 10.3389/fphys.2019.01044

Deep Learning-Based Structure-Activity Relationship Modeling for Multi-Category Toxicity Classification: A Case Study of 10K Tox21 Chemicals With High-Throughput Cell-Based Androgen Receptor Bioassay Data

Gabriel Idakwo[1]†, Sundar Thangapandian[2]†, Joseph Luttrell IV[1], Zhaoxian Zhou[1], Chaoyang Zhang[1]* and Ping Gong[2]*

[1] School of Computing Sciences and Computer Engineering, The University of Southern Mississippi, Hattiesburg, MS, United States, [2] Environmental Laboratory, U.S. Army Engineer Research and Development Center, Vicksburg, MS, United States

Deep learning (DL) has attracted the attention of computational toxicologists as it offers a potentially greater power for *in silico* predictive toxicology than existing shallow learning algorithms. However, contradicting reports have been documented. To further explore the advantages of DL over shallow learning, we conducted this case study using two cell-based androgen receptor (AR) activity datasets with 10K chemicals generated from the Tox21 program. A nested double-loop cross-validation approach was adopted along with a stratified sampling strategy for partitioning chemicals of multiple AR activity classes (i.e., agonist, antagonist, inactive, and inconclusive) at the same distribution rates amongst the training, validation and test subsets. Deep neural networks (DNN) and random forest (RF), representing deep and shallow learning algorithms, respectively, were chosen to carry out structure-activity relationship-based chemical toxicity prediction. Results suggest that DNN significantly outperformed RF ($p < 0.001$, ANOVA) by 22–27% for four metrics (precision, recall, F-measure, and AUPRC) and by 11% for another (AUROC). Further in-depth analyses of chemical scaffolding shed insights on structural alerts for AR agonists/antagonists and inactive/inconclusive compounds, which may aid in future drug discovery and improvement of toxicity prediction modeling.

Keywords: deep neural networks, deep learning, random forest, androgen receptor, structure-activity relationship, multi-class classification, agonist, antagonist

Edited by:
Ioannis P. Androulakis,
Rutgers, The State University of New Jersey, United States

Reviewed by:
Alessandro Giuliani,
Istituto Superiore di Sanità (ISS), Italy
Marianthi Ierapetritou,
University of Delaware, United States

****Correspondence:***
Chaoyang Zhang
Chaoyang.Zhang@usm.edu
Ping Gong
Ping.Gong@usace.army.mil

† These authors have contributed equally to this work

Specialty section:
This article was submitted to Systems Biology, a section of the journal Frontiers in Physiology

Received: *20 December 2018*
Accepted: *30 July 2019*
Published: *13 August 2019*

Citation:
Idakwo G, Thangapandian S, Luttrell J IV, Zhou Z, Zhang C and Gong P (2019) Deep Learning-Based Structure-Activity Relationship Modeling for Multi-Category Toxicity Classification: A Case Study of 10K Tox21 Chemicals With High-Throughput Cell-Based Androgen Receptor Bioassay Data. Front. Physiol. 10:1044. doi: 10.3389/fphys.2019.01044

INTRODUCTION

Toxicity caused by chemical exposure can be manifested sequentially at ascending organismal levels, which often begins as a molecular initiating event and escalates into adverse effects measured as toxicological endpoints for the cell, tissue, organ, organism, or population (Ankley et al., 2010; Organization for Economic Co-operation and Development [OECD], 2013; Allen et al., 2014). There exist three categories of chemical toxicity testing strategies: *in vivo*, *in vitro,* and *in silico*. Due to the prohibitively high costs and ethical concerns over animal welfare associated with *in vitro* and *in vivo* assays, there has been an increasing demand for reduced animal use as well as a shift in toxicity testing paradigms from *in vivo*/*vitro* to *in silico* (National Research Council, 2007). This demand has also been driven by the 3Rs (Replacement, Reduction, Refinement) movement (Stokes, 2015) and by government policies, regulations and legislation [e.g., REACH by the European Union (2006)]. Despite significant advances made in the past decades, *in silico* prediction of chemical toxicity without performing any biochemical (ligand binding) or *in vitro*/*vivo* assays remains an unresolved challenge (Li et al., 2018). Among all *in silico* approaches, structure-activity relationship (SAR)-based modeling has become the predominant one, and it is capable of both qualitative classification and quantitative prediction.

Once the toxicity endpoint or biological activity for prediction is set, the performance of SAR-based predictive modeling is largely determined by the choice of molecular descriptors relevant to toxicity (Shao et al., 2013) and of the prediction modeling algorithms (Plewczynski et al., 2006). The latter varies from linear methods, such as multiple linear regression (MLR), partial least squares (PLS), and linear discriminant analysis (LDA) to non-linear methods, such as *k*-nearest neighbors (KNN), artificial neural networks (ANN), decision trees, and support vector machines (SVM) (Dudek et al., 2006). Recently, deep learning (DL), with the Rectified Linear Unit (ReLU) activation function and such architectures as recurrent neural networks (RNN) and convolutional neural networks (CNN), has emerged as a promising tool for *in silico* toxicity or bioactivity prediction modeling (Hughes et al., 2015, 2016; Xu et al., 2015; Gao et al., 2017; Hughes and Swamidass, 2017; Wu and Wang, 2018). DL, also called deep structured learning or hierarchical learning, allows computational models that are composed of multiple processing layers to be fed with raw data and automatically learn multiple levels of abstract representations of data for performing detection and classification (LeCun et al., 2015). The success of DL has been well documented in such diverse fields as image and speech recognition (Shen et al., 2017; Cummins et al., 2018), visual art (Huang et al., 2016c), natural language processing (Névéol et al., 2018), drug discovery (Dana et al., 2018), bioinformatics (Min et al., 2016), computational biology (Angermueller et al., 2016), and the game of GO (AlphaGo) (Silver et al., 2016).

One of the earliest case studies of applying DL in SAR-based toxicity prediction was reported by Mayr et al. (2016) who developed the DeepTox pipeline. The authors trained deep neural networks (DNNs) using the Tox21 Data Challenge dataset (i.e., training data) that consisted of approximately 12,000 compounds and 12 *in vitro* bioassays (Huang et al., 2016a; Huang and Xia, 2017), and then they predicted the toxicity of approximately 650 chemicals (test data). Although the multi-task DNN exceled in terms of the average AUC (Area Under the Curve of receiver operating characteristics) of the overall 12 bioassays, the nuclear receptor (NR) signaling panel (7 assays), and the stress response (SR) panel (5 assays), it did not perform as well for 5 out of the 12 bioassays as conventional shallow learning techniques did (e.g., SVM, random forest (RF), and elastic net) (Mayr et al., 2016). These results are consistent with the performance of DeepTox in the Tox21 Data Challenge competition where the DeepTox pipeline ranked behind several shallow learning techniques for half of the 12 bioassays even though it won 9 sub-challenges, including those for the other 6 bioassays, the NR and the SR panels, and for the 12 bioassays overall (Mayr et al., 2016; Huang et al., 2016a).

In the past 3 years, more than a dozen papers have been published with conflicting conclusions on comparative performance between DL and shallow learning. For instance, the deepAOT (DL-based acute oral toxicity) models constructed using a molecular graph encoding convolutional neural network (MGE-CNN) architecture outperformed previously reported shallow learning models in both quantitative toxicity prediction and toxicant category classification (Xu et al., 2017). By pairing element specific topological descriptors (ESTDs) with multitask DNN, TopTox (topology-based multitask DNNs) was demonstrated to be more accurate than RF and gradient boosting decision tree (GBDT) using four benchmark ecotoxicity datasets (Wu and Wei, 2018). On the contrary, SVM outperformed DNN in predictive classification of chemical-induced hepatocellular hypertrophy (Ambe et al., 2018), and multiple layer perceptron (MLP) exceeded the performance of 2D ConvNet (2D Convolutional neural network) in the aforementioned 12 Tox21 bioassays (Fernandez et al., 2018). Meanwhile, Liu et al. (2018) found that the overall performance of DNN models was similar to that of RF and variable nearest neighbor methods. They also concluded that neither a larger number of hidden neurons nor a larger number of hidden layers necessarily leads to better neural networks for regression problems. This contradicted previous observations that deeper and wider networks generally performed better than shallower and narrower ones (Koutsoukas et al., 2017; Lenselink et al., 2017). Recently, Mayr et al. (2018) conducted a large-scale comparison of drug target prediction between DL (Feed-forward neural networks or FNN, CNN, and RNN) and shallow learning (RF, SVM, KNN, naïve Bayes (NB), and similarity ensemble approach) methods using a large benchmark dataset (456,331 compounds and more than 1000 assays) from the ChEMBL database. Although FNN was statistically identified as the frontrunner across a wide variety of assay targets, the authors observed that RF and SVM had higher average AUC scores than CNN and RNN.

As a new domain with less than 5 years of application history, we have yet to see overwhelmingly significant and convincingly consistent improvements in both quantitative prediction and qualitative classification of chemical toxicity using DL. Evidence has indicated that DL sometimes does enhance prediction accuracies over shallow learning. However, obtaining such results appears to occur on a case-by-case basis, and the opposite outcomes have also been reported. More studies are warranted to look into many confounding factors such as descriptors, assay targets, chemical space, hyper-parameters, and DL architectures, all of which may impact the performance of DL in QSAR-based chemical toxicity prediction.

Motivated by the aforementioned controversy, we conducted the present study to further investigate if DL algorithms could be optimized to offer a significant improvement over representative shallow learning algorithms for a suite of performance metrics. In the following, we first describe two Tox21 quantitative high throughput screening (qHTS) assay datasets with more than 10,000 compounds. These cell-based qHTS assays were conducted to identify small molecule agonists and antagonists of the androgen receptor (AR) signaling pathway (Huang et al., 2016b). Then, such structural features as 1D–3D molecular descriptors and fingerprints were computed for each chemical. Two algorithms, i.e., DNN (representing DL) and RF (representing shallow learning), were employed to build SAR-based classification models so as to compare the accuracy of these methods for predicting chemical class labels (i.e., agonist, antagonist, inactive, and inconclusive). Our results suggest that DNN outperformed RF not only significantly by statistical analysis, but by a large margin of more than 20% in four of the five performance metrics. Further in-depth analyses of chemical scaffolding shed insights on the structural alerts for the four classes of chemicals in AR activity, which may aid in future drug discovery and improvement of toxicity prediction modeling.

MATERIALS AND METHODS

Bioassay Dataset Curation and Preprocessing

Toxicology in the 21st century (Tox21) is a collaborative initiative launched by the consortium of the NIH, EPA and FDA aiming to develop better toxicity assessment methods[1]. The Tox21 program has tested over 10,000 chemicals against a panel of NR and SR signaling pathways (Attene-Ramos et al., 2013; Huang et al., 2016b). AR, a nuclear hormone receptor, plays a critical role in AR-dependent prostate cancer and other androgen related diseases (Tan et al., 2015). Two *in vitro* assays were carried out in both agonist mode and antagonist mode to assess the agonistic and antagonistic properties of Tox21 chemicals, respectively. The first assay (BLA assay) used the AR-UAS-bla-GripTite^{TM} cell line that contained the ligand-binding domain (LBD) of the rat AR and stably expressed a beta-lactamase reporter gene under the transcriptional control of an upstream activator sequence (UAS). The second assay (MDA assay) used a human breast carcinoma cell line (MDA-kb2 AR-luc) stably transfected with a luciferase reporter gene. A total of 10,496 chemicals were tested, and their assay outcomes were downloaded from the Tox21 Data Challenge website[2]. The downloaded datasets (2 assay modes × 2 assays) were merged using PubChem Substance IDs (SID) because SID was unique for each entry in the datasets. Of the 10,496 compounds, 149 compounds were mixtures of chemicals such as oils and solvents and another 96 compounds contained atoms for which reliable force field parameters were unavailable to perform molecular docking with (see section "Chemical Structure Preparation" below). Thus, these 245 compounds were removed. There was redundancy in the remaining compounds because, on some occasions, multiple SIDs were found corresponding to the same PubChem Compound ID (CID). Hence, CIDs were used to identify and remove redundant chemicals, resulting in 7665 unique chemicals (see **Supplementary Figure S1**).

For each SID entry, there were up to four records of qualitative assay outcomes that resulted from two assays (BLA and MDA) in two assay modes (agonist and antagonist). There were three possible assay outcomes, i.e., active agonist, active antagonist, or inactive. We assigned one of four class labels, namely "agonist," "antagonist," "inactive," or "inconclusive," to each chemical by adopting the following rules: a chemical was labeled (i) "agonist" only if both assays in the agonist mode determined it to be an active agonist, (ii) "antagonist" only if both assays in the antagonist mode determined it to be an active antagonist, (iii) "inactive" if all assay outcomes for this chemical were negative, or (iv) "inconclusive" if any other combination was true. In the case of chemical entry redundancy, i.e., multiple SIDs corresponding to the same CID, a consensus was reached on the class label by selecting the most frequently occurring response (i.e., the assay outcome with the highest incidence of occurrence), or the chemical was removed if the assay outcomes were evenly split among multiple categories. Finally, 7665 unique chemicals with unambiguous consensus assay outcomes were obtained and used in the downstream steps (see **Supplementary Figure S1**).

Chemical Dataset Curation and Preprocessing

Chemical Structure Preparation

The SMILES of the 7665 unique chemicals were downloaded from PubChem via its PUG REST interface[3] (Kim et al., 2018) using a custom R script. The Open Babel program (O'Boyle et al., 2011) was used to perform the following steps to clean and optimize the downloaded chemical structures (also see **Supplementary Figure S1**). Salts and other small fragments were removed and only the largest fragment of each entry was retained. SMILES were converted to 2D structures and hydrogens were added when necessary. Then, 3D conformations were generated and partial charges were assigned using the *Electronegativity Equalization Method* followed by energy minimization using the *steepest descent* algorithm (Bultinck et al., 2002; Geidl et al., 2015). Finally, molecular docking was performed to generate biologically relevant 3D ligand conformations within the binding site of the AR because the bound ligand conformation was typically different from the

[1] https://ncats.nih.gov/tox21/about/goals

[2] https://tripod.nih.gov/tox21

[3] https://pubchemdocs.ncbi.nlm.nih.gov/pug-rest

conformations obtained in its unbound state (Tirado-Rives and Jorgensen, 2006; Thangapandian et al., 2010). Molecular docking was performed using the AutoDock Vina program (Trott and Olson, 2010) and the X-ray crystal structure of AR-testosterone complex (PDB ID. 2AM9) (de Jésus-Tran Karine et al., 2006). A cubic box of 16 Å × 16 Å × 16 Å centered at the binding site was used to dock the chemicals in the data set. The docking-generated ligand conformations were used for 3D descriptor calculations (see section "Feature Generation and Dimensionality Reduction" below).

Feature Generation and Dimensionality Reduction

A total of 17,967 molecular descriptors and fingerprints (termed features) were generated using PaDEL (Yap, 2011), including 1444 1D or 2D descriptors, 431 3D descriptors, and 16,092 unique fingerprints belonging to 12 different pattern types. The 3D descriptors were calculated using the binding conformations obtained above from molecular docking. In case PaDel failed to compute certain features for certain compounds, the mean-imputation method as implemented in Scikit-Learn (Pedregosa et al., 2011) was employed to replace those missing values. A variance thresholding method was used to reduce feature dimensionality. Any feature vector with at least 85% of its entries being identical was removed, resulting in a final set of 2544 features.

Feature Standardization

For many algorithms, it is necessary to rescale the features to keep certain features from getting more influence than they should. This particularly holds true for neural networks where certain weights may update faster than others, thus making optimization methods converge less quickly (LeCun et al., 2015). Also, the generated features were of varying scales and distributions, and they were also comprised of count and binary features. To resolve this, the features in the final set were standardized (rescaled) individually such that they assumed a standard normal distribution with a mean of zero and unit standard deviation. Using the StandardScaler function in Scikit–Learn (Pedregosa et al., 2011), the training dataset was rescaled by subtracting the mean and dividing the resulting difference by the standard deviation. The mean and standard deviation used in the training dataset were used to transform the test dataset.

Chemical Space Visualization

The chemical space of the 7665 unique Tox21 chemicals was visualized in two-dimensional vectors. The space of the final set of 2544 features was further reduced to two abstract features using an autoencoder (Baldi, 2012; Chandra and Sharma, 2015). By trying to reconstruct the input at the output layer, the autoencoder was forced to learn the underlying feature space in a lower dimension. The innermost layer of the autoencoder, an embedding of the input, was set to two units. The encoder component of the autoencoder had 2544 units in the input layer corresponding to the number of features in the input data and {1024, 512, 128, 32, 2} features in the hidden layers. The decoder component of the autoencoder was ordered as the reverse of the encoder. For activation functions, ReLU was used in the hidden layers while sigmoid functions were used in the output layer. The Adam optimizer was used to minimize the mean squared error. The autoencoder model was trained using the Keras (Chollet, 2015) Python library with a Tensorflow backend.

Machine Learning Methods

Machine Learning-Based SAR Modeling Approach

The overall workflow of our machine learning-based SAR modeling approach is illustrated in **Figure 1**. It began with data curation, followed by preprocessing of chemical structure and *in vitro* assay data. We employed a nested double-loop cross-validation strategy to ensure robust model development and to alleviate the impact of selection bias and overfitting (Cawley and Talbot, 2010). Similar to most other typical SAR datasets, the 7665 unique chemicals displayed an imbalanced distribution across the four assay outcome classes, i.e., agonist, antagonist, inactive, and inconclusive. As a result of the imbalance, a stratified sampling strategy was adopted to ensure that the partitioning of chemicals across all classes remained the same between the cross-validation folds and between the training and test datasets.

The 7665 chemicals were split randomly using the stratified strategy into five subsets. For each run of the outer loop, one subset (20%) was withheld as the test set while the remaining four subsets (80%) were used as the training set. Each of the five runs in the outer loop used a different subset. In the inner loop, the training set was further randomly split into 10-folds using the stratified strategy. Ninefolds were used for model (classifier) training or hyper-parameter tuning, while the remaining onefold was used for validation. Thus, a 10-fold cross-validation was implemented in the inner loop for classifier training, whereas a fivefold cross-validation was executed in the outer loop for model testing and evaluation. The overall performance was assessed using the average metrics values of all five runs in the outer loop (see section "Chemical Scaffolding and Similarity Analysis" for metrics definition).

Shallow and Deep Learning Algorithms

Six commonly used and popular machine learning algorithms were compared in a preliminary study. They included KNN, RF, classification and regression trees (CART), NB, SVM, and DNN, all of which ran under their respective default settings as implemented in Scikit-Learn (Pedregosa et al., 2011). Their performance without optimization was determined by following the workflow presented in **Figure 1**. Based on their performance metrics as shown in **Supplementary Figure S2**, we selected the top two algorithms, DNN and RF, for further optimization and chemical toxicity classification in this study.

Random forest and optimization

Random forests are a collection of decision trees whose predictions are averaged to obtain an ensemble performance. Randomness is achieved by allowing each tree in the forest to use bootstrap samples of the training data and random molecular features selection for prediction. Decision Trees are drawn upside down and begin with a trunk that splits into multiple branches before eventually arriving at the leaves. The leaf nodes represent the endpoint to be predicted, while all other

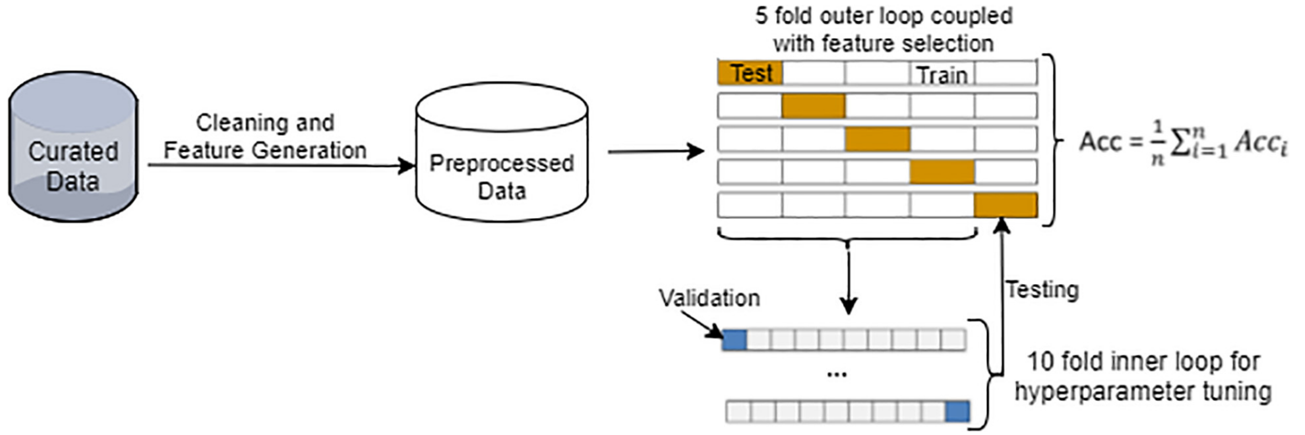

Curated Data
Cleaning and Feature Generation
Preprocessed Data
5 fold outer loop coupled with feature selection
Test
Train
$Acc = \frac{1}{n}\sum_{i=1}^{n} Acc_i$
Validation
Testing
...
10 fold inner loop for hyperparameter tuning

FIGURE 1 | Overview of the machine learning-based SAR approach with a nested double-loop cross-validation strategy for model construction, validation, and evaluation. Details of data preprocessing are provided in **Supplementary Figure S1**.

nodes are assigned a molecular feature. To construct a robust decision tree, the features (nodes) that most clearly differentiate the endpoints (leaf nodes) are chosen. *Gridsearch* with 10-fold cross validation was employed in optimizing the RF models. The distribution of parameters optimized for the RF model is provided in **Supplementary Table S1**.

Deep learning and optimization

Deep learning architecture. We briefly describe this algorithm and the hyper-parameters of DNNs in order to facilitate our discussion of the optimization and performance analysis process. A DNN is an artificial neural network with one input layer, multiple hidden layers and one output layer, as shown in **Figure 2**. The number of hidden layers is defined as k. Each layer consists of a number of units (or neurons), denoted by n. The number of units at the input layer corresponds to the number of features in the input data (x). The number of units in the output layers is equal to the number of classes to be predicted. In this study, there were four units in the output layer that corresponded to four classes: (i) agonist, (ii) antagonist, (iii) inactive, and (iv) inconclusive. The number of units in each hidden layer usually depends on specific details of various classification problems and datasets. Typically, it is determined by multiple trials of different network topologies. For a fully connected network as used for this study, each pair of units i and j in two consecutive layers are connected by a link with a weight $W_{i,j}$. There is an input and an output for each unit. In the input layer, the output is the same as the input for each unit. For each unit in the hidden layer, the input is comprised of the weighted sum of the units in the previous layers and the bias of the current unit. The output of each hidden layer unit is obtained by applying an activation function to its input. The ReLU activation function is applied to all units in all the hidden layers and computes the function $f(x) = \max(0,x)$. This allows for easy gradient computation, which in turn results in faster training for large networks. By feeding the training data in batches to the input layer (with a specified batch size), the DNN with a given network topology and weights can compute the predictions in the output layer. During the training process, a dropout regularization technique is used to ignore some randomly selected neurons in order to prevent the neural networks from overfitting. Dropout rate is a parameter that needs to be tuned in DL. The softmax function is applied to the output layer to obtain a categorical probability distribution with values between 0 and 1, indicating the likelihood that any of the four classes are true. The highest probability determines the class label of each sample.

Learning process. Training a neural network with a given architecture is a process performed to find a combination of weights of units so as to minimize the error between the predictions in the output layer and the known truth. In our study, categorical cross entropy θ is used as the loss function to compute the error. We can minimize the objective function θ by iteratively applying optimization methods such as mini-batch gradient descent, Adam, RMSprop, and Adagrad. Backpropagation is used in gradient descent methods to update the weights of units by computing the gradient $\nabla\theta$ of the loss function with respect to weight $W_{i,j}$.

The weights are updated in the opposite direction of $\nabla\theta$. The update of the weight $w_{i,j}$ is defined as $\Delta w_{i,j} = -l\frac{\partial\theta}{\partial w_{i,j}}$

where l refers to the learning rate that determines the size of the steps taken at each iteration to reach the minimum of the objective function. The weights are updated iteratively, and the learning process repeats until the neural networks are trained adequately. This means that the loss function decreases to a certain threshold.

Hyper-parameter optimization. The hyper-parameters in DL need to be tuned to get the best model suited for the dataset. These hyper-parameters include the number of hidden layers, the number of units in the input layer, the number of units in the hidden layers, the number of units in the output layer (e.g., set to 4 in this study because of the four categories of the chemical activity classification), batch size, dropout rate, learning rate and optimizer.

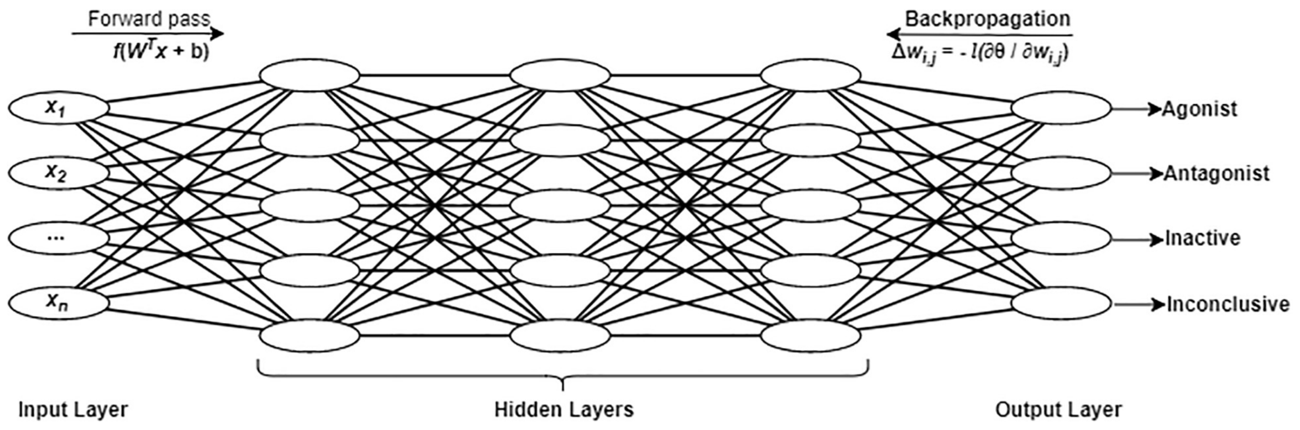


FIGURE 2 | A fully connected deep neural network with an input layer, three hidden layers, and an output layer. As an example, four units are shown in the output layer corresponding to the four chemical activity classes in the present study.

Bayesian hyper-parameter optimization has been shown to perform faster and more accurately than grid and random parameter search, respectively (Snoek et al., 2012). The rationale for Bayesian optimization is to liken the optimization of hyper-parameters to a function minimization challenge. In Bayesian hyper-parameter optimization, a probability model of the objective function is constructed, which is often referred to as a surrogate function and denoted as p(score | parameters). Instead of randomly selecting parameters or going through a grid in a blind manner, the results of the surrogate function are used to select the next parameters to try on the objective function, thus minimizing the number of calls to the objective function. The hyper-parameters with the best score or least validation set error computed by the objective function are considered the optimal. In this study, the search for optimal hyper-parameters was conducted using Bayesian optimization as implemented in Hyperas, a tool that combines the Keras DL library (Chollet, 2015) with Hyperopt's Sequential Model-Based Optimization (SMBO) methods using the Tree-structured Parzen Estimator (TPE) algorithm (Bergstra et al., 2011). The search space included hidden layers {2,3,4}, Neurons {32,64,128,256,512,1024}, optimization methods {mini-batch gradient descent, Adam, RMSprop, Adagrad}, batch size {8,16,32,64,128}, and learning rate {random uniform distribution between 0 and 1}.

Model Evaluation Metrics

Five metrics were computed for model performance evaluation. They included precision, recall, F1-score (also called F-measure), the area under the receiver operating characteristic curve (AUROC), and the area under the precision-recall curve (AURPC). Macro-averages of the performance metrics were calculated and used for evaluation throughout this study because of the imbalanced nature of the data and the multi-category classification task. Macro-averaging independently computes the average for every class prior to averaging. By giving the same weight to all classes, it can show how effective a model is on the minority classes, e.g., AR agonists and AR antagonists that are of greater importance in this study. Micro-averaging was not considered as it gives equal weight to every sample; hence, the majority classes contribute more to the average metric than the minority classes. The following formulas describe computing the macro-averages of precision, recall and F-measure.

$$\text{Precision}_{\text{macro}} = \frac{\sum_{i=1}^{m} \frac{tp_i}{tp_i+fp_i}}{m}$$

$$\text{Recall}_{\text{macro}} = \frac{\sum_{i=1}^{m} \frac{tp_i}{tp_i+fn_i}}{m}$$

$$\text{F} - \text{measure}_{\text{macro}} = \frac{\sum_{i=1}^{m} \left(\frac{2\times\text{Precision}_i\times\text{Recall}_i}{\text{Precision}_i+\text{Recall}_i}\right)}{m}$$

where m = number of classes, tp = true positive, fp = false positive, fn = false negative.

The AUROC and the AUPRC were determined in Scikit–Learn (Pedregosa et al., 2011) by computing the area under the plot of true positive rate vs. false positive rate and that of precision vs. recall, respectively. The macro-averages of AUROC and AUPRC were calculated in a similar fashion to those of precision and recall above.

Implementation Environment

The machine learning models were developed in Python 3.5.4 using Jupyter Notebook within the Anaconda 4.3.27 (64-bit) environment. Other important libraries include Scikit-Learn 0.19.0, Keras 2.1.4, Tensorflow 1.9, and Hyperas 0.4. All models were trained on a server (Intel Xeon E5-1650) running Ubuntu 16.04.5 LTS with six cores, 32GB memory and four Nvidia Titan Xp GPUs.

Chemical Scaffolding and Similarity Analysis

Chemical scaffolding and similarity analysis were performed on one of the five chemical subsets used as the external test set in the first run (i.e., Fold 1 as seen in **Figure 1** and **Supplementary Table S2**). The R packages *Rcdk* and *Rcpi* were used for calculating chemical scaffolds and similarity analysis, respectively. The true labels (not predicted labels) of chemicals were used for both analyses.

In chemical scaffolding, the structural information of a chemical can be organized into rings and frameworks (Bemis and Murcko, 1996). Any cycles that share an edge are defined as rings, whereas any unions of rings via linkers are defined as frameworks. For instance, benzene, naphthalene, and anthracene are single ring systems, whereas diphenylmethane is a framework. Using Murcko chemical scaffolding, a list of rings and frameworks present in the test chemicals was generated.

The Tanimoto coefficient or scores (Bajusz et al., 2015) are a widely accepted metric for evaluating similarity between two chemicals. We calculated the Tanimoto scores, using the PubChem fingerprints as the input, for every interclass pairing (e.g., an agonist vs. an antagonist, an agonist vs. an inactive, an antagonist vs. an inconclusive) in order to compare interclass similarity. The score of 0.5 was selected as the cutoff threshold, i.e., any pairs of chemicals with a score $\geq$ 0.5 were considered similar to each other.

RESULTS AND DISCUSSION

Data Distribution and Evaluation Metrics

As shown in **Figure 3A**, the 7665 unique compounds were unevenly distributed across four AR activity classes with the two active classes (222 compounds) being the minority (2.9%) and the inactive (2476) or inconclusive (4967) classes being the majority (97.1%).

An autoencoder was used to reduce chemical feature dimensionality. As a result, the chemical space distribution of the final set of 7665 compounds can be visualized in a 2D plot (**Figure 3B**). The plot shows that no class forms a distinct cluster, the two inactive classes are more widely dispersed than the two active classes, and that all the active compounds reside within the space of inactive or inconclusive ones. These observations suggest that it was a challenging task to separate the four classes based on the structural features of the compounds.

Owing to the skewed class distribution, one of our main objectives was to develop a classification model with high performance for the minority classes because the two less populated active classes were of higher toxicological importance. Meanwhile, the model should not sacrifice the accuracy of the more abundant inactive and inconclusive classes, which would compromise the overall prediction performance for the entire dataset. Therefore, we chose to use macro-averages over micro-averages (see section "Model Evaluation Metrics" above) and selected evaluation metrics that are sensitive to class imbalance or favorable to minority classes such as F-measure and AUPRC (Jeni et al., 2013). F-measure is considered a better metric than precision (P) and recall (R) because it is a harmonic mean of P and R and also a tradeoff between P and R (Powers, 2011). Although AUROC and AUPRC both provide model-wide evaluation, a classifier that optimizes the area under ROC is not guaranteed to result in an optimal AUPRC (Davis and Goadrich, 2006). When the positives are the minority and more important than the negatives, AUROC is an overly optimistic measure of model performance, whereas AUPRC provides a more informative and accurate depiction of model prediction performance as it evaluates the fraction of true positives among positive predictions (Saito et al., 2015).

Performance Comparison Between DNN and RF

Only F-measure was determined in the preliminary performance study of six machine learning algorithms without parameter optimization, and RF showed the highest F-measure with a low variance (**Supplementary Figure S2**). Therefore, RF was selected to represent shallow learning algorithms for further optimization as well as to compare with DNN.

Following the workflow depicted in **Figure 1**, we optimized the hyper-parameters, built multi-class prediction models, and assessed the model performance. Details of the hyper-parameter optimization approach for RF and DNN are described earlier in section "Shallow and Deep Learning Algorithms." The optimized parameters for RF are provided in **Supplementary Table S1**. For DNN, we found that (a) the architecture of the best performing classifier had three hidden layers with (1024,1024,512) units; (b) regularization was achieved using dropout rates of (0.25, 0.341, and 0.5) applied on these three hidden layers, respectively; and (c) Mini-Batch Gradient Descent with a batch size of 16 allowed for frequent updates in the weights of the network and a more robust convergence.

Then, DNN and RF models were separately trained using the same preprocessed data. **Figures 4A,B** present the confusion matrices and the average recall scores for all four classes calculated from the external fivefold cross-validation (see **Supplementary Tables S2–S6** for detailed reports for folds 1–5, respectively). **Figure 4C** provides the average performance metrics for DNN and RF side-by-side (see **Supplementary Tables S7, S8** for the raw metrics data for all fivefolds). These results clearly indicate that DNN consistently outperformed RF in both of the following measures: (1) the average number of correctly classified compounds (recall) for all four classes (**Figures 4A,B**), and (2) the macro-averages of all five performance metrics across all four classes (**Figure 4C**).

Specifically, DNN correctly predicted 50% more antagonists and 28% more inconclusive compounds than RF did, whereas the other two classes were not improved as much (i.e., 18% for agonists and 7% for inactive compounds) (**Figures 4A,B**). Furthermore, the performance enhancement was statistically significant ($p < 0.001$, ANOVA) for each metric (**Figure 4C**), regardless of whether the metric is insensitive (AUROC) or sensitive (the other four metrics) to imbalanced class distribution (Jeni et al., 2013). It is worth noting that the four imbalance-sensitive metrics were improved by 22–27%, while AUROC was boosted by only 11%. The coefficient of variation (CV = standard deviation/mean) for each metric was less than 5% except for the precision of RF (17%), suggesting that both DNN and RF models had stable performance (**Supplementary Tables S7, S8**). However, the performance of DNN models was

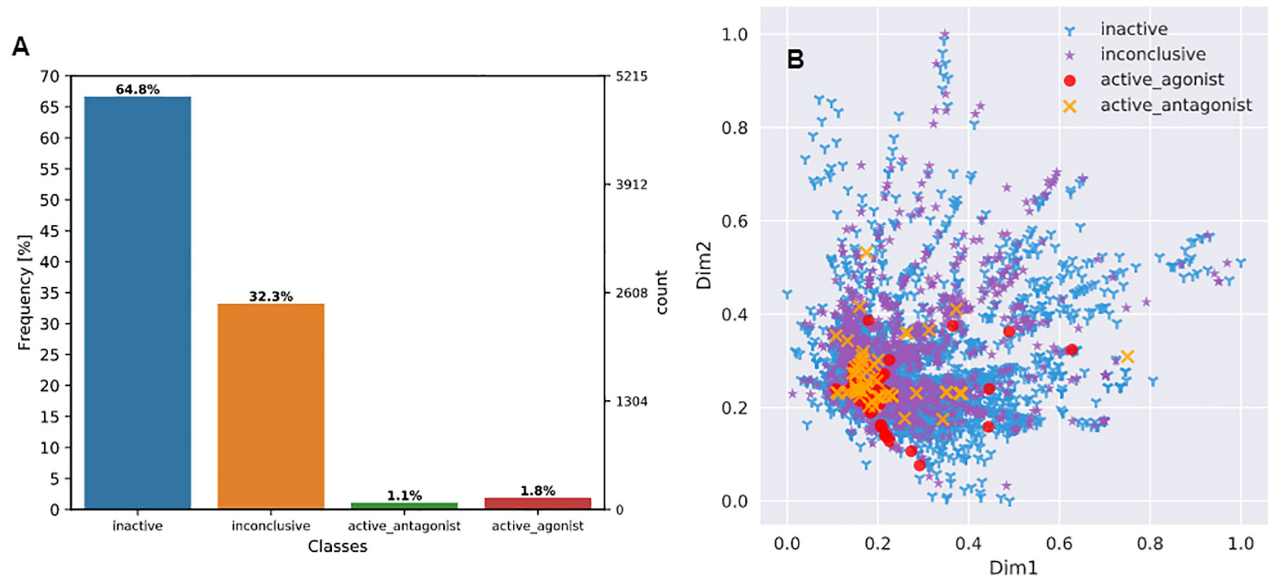


FIGURE 3 | Visualization of chemical distribution over activity classes and chemical space. **(A)** Distribution of Tox21 compounds across four AR activity classes; **(B)** Distribution of all four AR activity classes of compounds over the compressed 2-dimension chemical space. Feature dimensionality was reduced from 2544 to 2 using an autoencoder (see section "Chemical Space Visualization" for more details).

more stable than that of RF (as reflected by much smaller CVs shown in **Supplementary Tables S7, S8** and lower error bars seen in **Figure 4C**).

However, performance did not differ between RF and DNN prior to hyper-parameter optimization in terms of F-measure: 0.548 ± 0.038 for RF vs. 0.536 ± 0.052 for DNN (p = 0.654, paired t-test; see **Supplementary Figure S2**). Parameter optimization did not enhance RF performance (F-measure): 0.548 ± 0.038 pre-optimization (**Supplementary Figure S2**) vs. 0.564 ± 0.029 post-optimization (**Figure 4C** and **Supplementary Table S8**) (p = 0.579, paired t-test). This was due to the fact that the default parameters for RF in Scikit–Learn were not arbitrary (i.e., they are pre-optimized for normal tasks) and were similar or comparable to the selected optimal ones (see **Supplementary Table S1**). On the contrary, hyper-parameter tuning greatly contributed to the improvement of DNN performance as reflected in the F-measure: 0.536 ± 0.052 pre-optimization (**Supplementary Figure S2**) vs. 0.832 ± 0.018 post-optimization (**Figure 4C** and **Supplementary Table S7**) (p < 0.001, paired t-test). It has come to our attention that some studies (e.g., Ambe et al., 2018; Fernandez et al., 2018) where suboptimal performance of DL was reported in comparison with shallow learning did not conduct adequate hyper-parameter optimization. These studies along with our own demonstrate the dependence of DL performance on hyper-parameter optimization.

Chemical Scaffolding Analysis

Using the chemicals in Fold 1 (20% of the entire preprocessed dataset) as an example, we conducted scaffolding analysis. Class-wise Murcko decomposition has revealed that the majority of chemicals contain single-ring systems and no Murcko frameworks (**Supplementary Figure S3**). Only 2 out of 28 agonists and 3 out of 17 antagonists contain scaffolding systems with more than one ring. These single-ring systems predominantly contain cyclopentanophenanthrene, a fused 4-membered ring system like in testosterone. About 20–30% inactive and inconclusive compounds contain systems with 2–4 rings (**Supplementary Figure S3A**). Both agonists and antagonists displayed a maximum of only three frameworks, whereas inactive and inconclusive compounds contained as many as 16 frameworks. This meant that the AR active compounds were more compact than the other two classes (**Supplementary Figure S3B**).

The obtained scaffolds (both rings and frameworks) were compared to explain the differences in prediction accuracy between different classes. The decomposed Murcko rings and frameworks revealed the total and unique chemical backbones present in each class (**Table 1**) as well as the class-specific backbones and those shared between classes (**Figure 5**). We identified 8 and 3 class-specific rings for AR agonists and antagonists, respectively (**Figure 5A**), as well as four frameworks unique to these two AR active classes (**Figure 5B**). Among the 4 agonist-specific frameworks, the 1,3-dioxole (a five-membered heterocycle consisting of two oxygen atoms at the 1 and 3 positions) and thiozetoquinoline (quinoline fused to a four-membered 1,3-thiazetidine) rings are each present in two frameworks, whereas piperazine (a six-membered ring containing two nitrogen atoms at para positions in the ring) is present in three frameworks (**Figure 6A**). A higher structural diversity is displayed in the antagonist-exclusive frameworks, including N-phenyl-azobicyclohexane-, naphthyridine-, piperidine-, and thiophene-containing frameworks, with only the structure of thiazole and piperidine connected by an ethyl

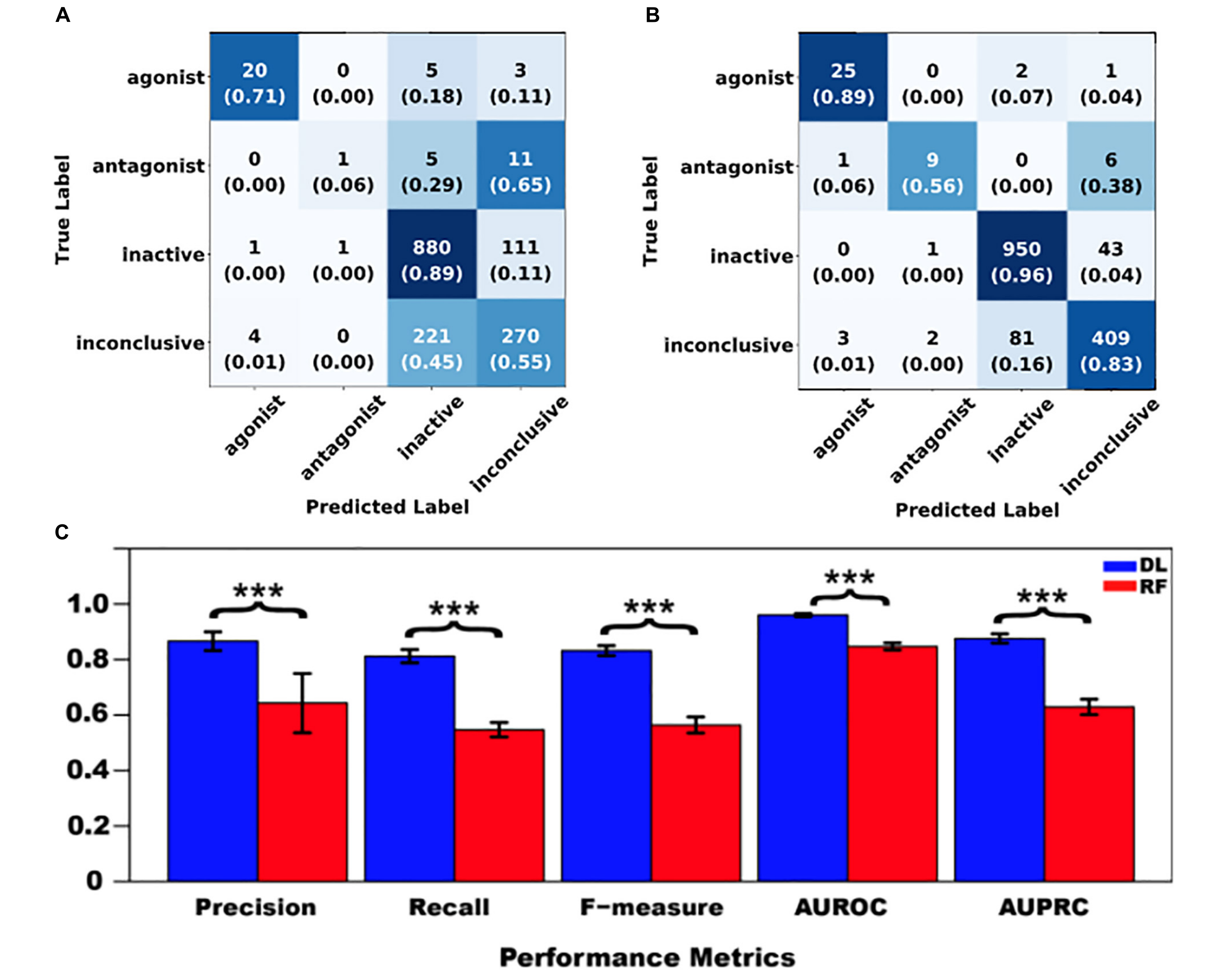


FIGURE 4 | Performance comparison between shallow learning algorithms represented by random forest (RF) and deep learning (DL) algorithms represented by deep neural networks (DNN). **(A)** RF confusion matrix; **(B)** DNN confusion matrix; and **(C)** Metrics comparison [mean ± standard deviation, $n = 5$; Here "***" stands for statistical significance at $p < 0.001$ (ANOVA, Tukey *post hoc* test)]. In confusion matrices, average numbers of predicted compounds and average recall scores (in parenthesis) for all four classes are shown, and all the cells are colored with a blue gradient (i.e., the darkness increases with the values).

linker present in two frameworks (**Figure 6B**). The 8 agonist- and 3 antagonist-specific rings are shown in **Figures 6C,D**, respectively. The low scaffold overlapping between agonists and antagonists (2 rings and 0 framework, **Figures 5A,B**) may explain why these two classes were rarely mistaken for each other during classification (**Figures 4A,B**). Furthermore, these class-specific scaffolds may serve as potential structural alerts for AR agonists or antagonists and as additional features in future machine learning-based classification or quantitative prediction modeling.

Among the four classes of chemicals, 65% (**Figure 4A**) vs. 38% (**Figure 4B**) of antagonists were misclassified as inconclusive compounds by RF and DNN, respectively; whereas 45% (**Figure 4A**) vs. 16% (**Figure 4B**) of inactive compounds were wrongly predicted to be inconclusive compounds by RF and DNN, respectively. These high rates of misclassification may be attributed to the high rates of non-redundant rings (5/9) and frameworks (2/6) present in antagonists that also appear in inconclusive compounds, and of non-redundant scaffolds (69/195 rings and 55/382

TABLE 1 | Numbers of total and non-redundant Murcko rings and frameworks present in the Fold 1 subset of Tox21 compounds.

	Rings		Frameworks	
Class	Total	Unique	Total	Unique
Agonist	30	14	4	4
Antagonist	20	9	7	6
Inactive	932	195	471	382
Inconclusive	648	167	611	497

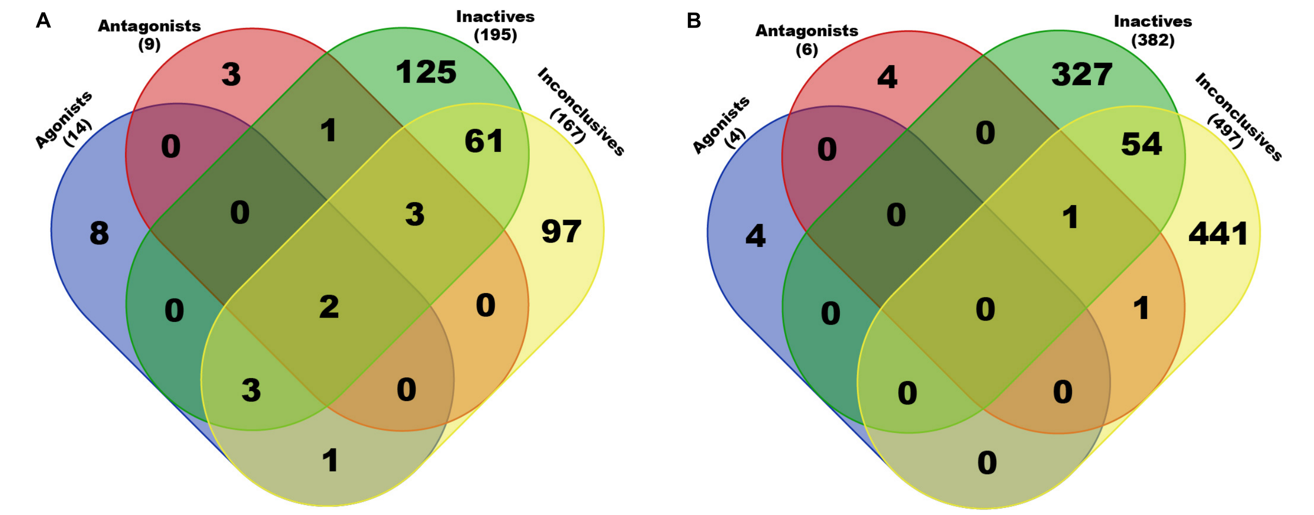


FIGURE 5 | Breakdown of exclusive and shared rings **(A)** and frameworks **(B)** present in each chemical class of AR activity. Only chemicals in the Fold 1 subset (20% of the final set of preprocessed compounds) were used in this analysis. Total numbers of non-redundant scaffolds are given in parentheses (also see **Table 1**).

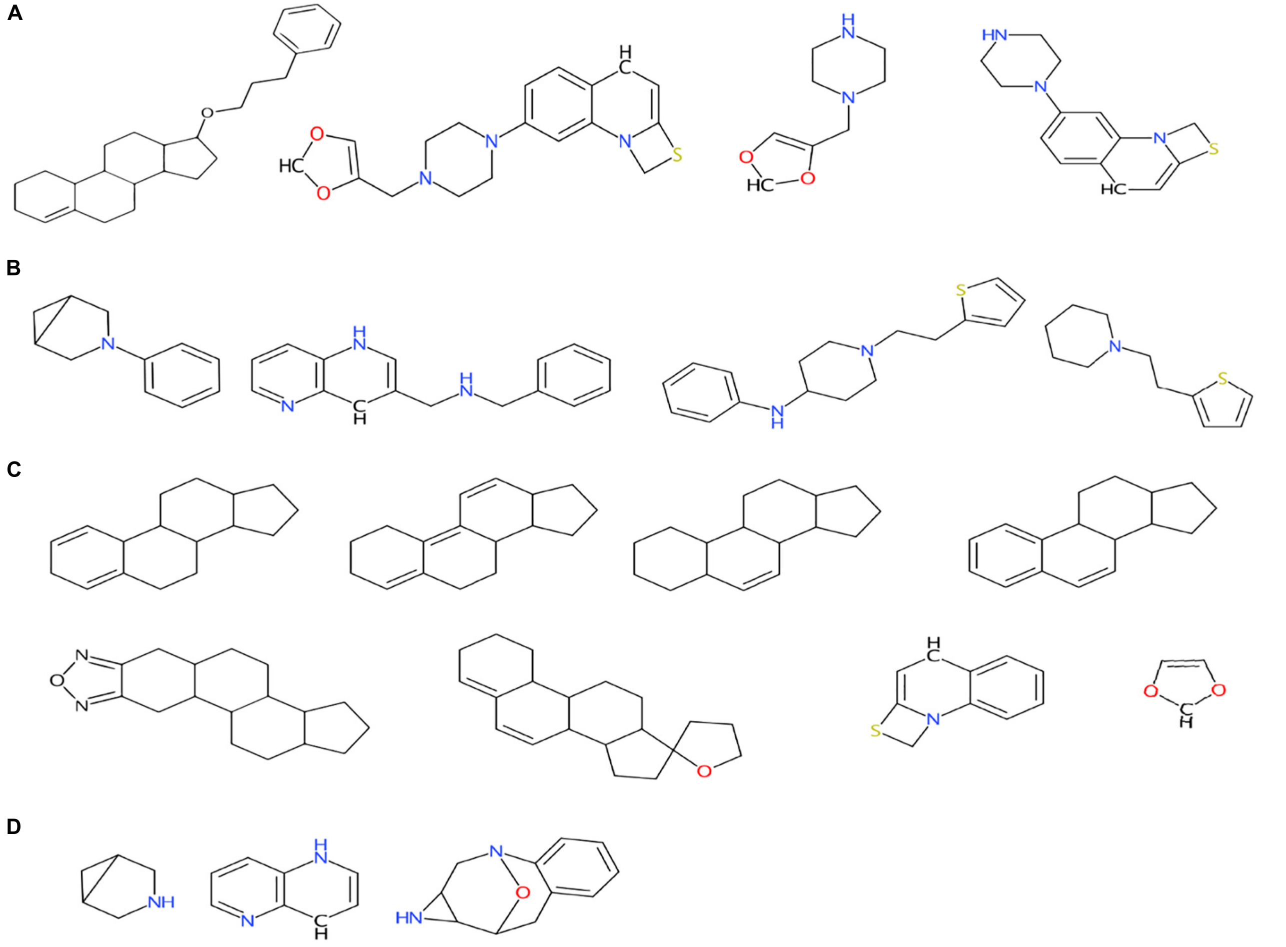


FIGURE 6 | Murcko frameworks exclusively present in agonists **(A)** and antagonists **(B)** as well as Murcko rings exclusively present in agonists **(C)** and antagonists **(D)**. Also see **Figure 5** for the numbers of class-specific frameworks and rings for these two classes.

frameworks) in inactive compounds overlapping with those in inconclusive compounds (**Figure 5**). For instance, the overlapping scaffolds between antagonist and inconclusive classes include five rings (benzene, pyrazoline, thiophene, piperidine and reduced cyclopenta[a]phenanthrene) (**Figure 7A**), and two frameworks (diphenylmethane and 4-phenylamino-piperidine) (**Figure 7B**). These overlapping scaffolds may confound the learning process in classification modeling, leading to lower prediction accuracies.

Chemical Similarity Analysis

The Tanimoto scores (TS) determined using PubChem fingerprints have revealed the degree of chemical similarity among the four AR activity classes. For the Fold-1 subset of Tox21 compounds, we determined five types of inter-class, pairwise chemical similarity: agonist-inactive, agonist-inconclusive, antagonist-inactive, antagonist-inconclusive, and agonist-antagonist (**Supplementary Figure S4**). It was observed that 4.1% (=1133/(28 × 994)) of agonist-inactive pairs and 4.0% (=544/(496 × 28)) of agonist-inconclusive pairs were chemically similar (TS ≥ 0.5), whereas 11.9% (=1788/(17 × 994)) of antagonist-inactive pairs and 10.5% (=875/(17 × 496)) of antagonist-inconclusive pairs were 50% or more similar (**Table 2**). Similar to scaffolding analysis results, the higher degree of chemical property similarity between antagonists and inconclusive or inactive compounds may have contributed to the high misclassification rates of antagonists (**Figures 4A,B**). In contrast, agonists, chemically less similar to inactive and inconclusive classes, were predicted with a much higher accuracy than antagonists (**Figures 4A,B**). The mean Tanimoto scores did not differ significantly among the four types of comparisons, likely due to an equalizing effect caused by high numbers of less similar chemical pairs (**Supplementary Figure S4**).

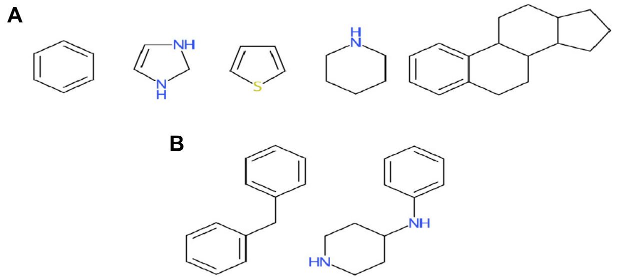


FIGURE 7 | Murcko rings **(A)** and frameworks **(B)** present in both antagonists and inconclusive compounds. Also see **Figure 5** for the breakdown of scaffolds among classes.

TABLE 2 | Pairwise Tanimoto scores (TS) between active and inactive/inconclusive classes in the Fold-1 subset of Tox21 compounds, consisting of 28 agonists, 17 antagonists, 994 inactive chemicals, and 496 inconclusive chemicals.

	Inactive (994)			**Inconclusive (496)**		
	# Pairs with TS ≥ 0.5	**Mean TS**	**% in all pairs**	**# Pairs with TS ≥ 0.5**	**Mean TS**	**% in all pairs**
Agonist (28)	1133	0.25 (±0.13)	4.1	544	0.29 (±0.13)	4.0
Antagonist (17)	1788	0.26 (±0.16)	11.9	875	0.31 (±0.17)	10.5

Shown here are the number of pairs with TS ≥ 0.5 and the percent of these pairs in the total number of possible pairs.

CONCLUSION

Using the multi-class AR dataset from the Tox21 Data Challenge, we conducted a case study to demonstrate that DL (represented by DNNs) was far superior to shallow learning (represented by RFs) for predicting their AR activities. Our results suggest that the performance of DNN was highly dependent on hyper-parameter optimization. Meanwhile, appropriate data preprocessing (e.g., feature generation and standardization), stratified data splitting, a double-loop cross-validation strategy and performance evaluation metrics also played an important role in ensuring high quality data, avoiding over-fitting, and alleviating the impact of skewed class distribution. By performing scaffolding and similarity analyses, we discovered potential causes for antagonists being frequently misclassified as inconclusive or inactive compounds and for inactive compounds being wrongly predicted as inconclusive compounds. The high similarity in chemical properties and structural scaffolding between antagonist and inconclusive compounds and between inactive and inconclusive compounds was identified as a confounding factor that impaired classifier performance. Meanwhile, a number of class-specific scaffolds have been identified as candidate structural alerts for AR agonists and antagonist, which may serve as additional chemical features to improve prediction performance in future studies.

AUTHOR CONTRIBUTIONS

PG and CZ conceived and supervised the study. GI and JL conducted the machine learning experiments. ST performed the data preprocessing, chemical scaffolding, and similarity analyses. ZZ and JL carried out the literature survey. All authors contributed to the manuscript writing and revision.

FUNDING

This work was supported by the U.S. Army Environmental Quality Research Program.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fphys.2019.01044/full#supplementary-material

REFERENCES

Allen, T. E. H., Goodman, J. M., Gutsell, S., and Russell, P. J. (2014). Defining molecular initiating events in the adverse outcome pathway framework for risk assessment. *Chem. Res. Toxicol.* 27, 2100–2112. doi: 10.1021/tx500345j

Ambe, K., Ishihara, K., Ochibe, T., Ohya, K., Tamura, S., Inoue, K., et al. (2018). In silico prediction of chemical-induced hepatocellular hypertrophy using molecular descriptors. *Toxicol. Sci.* 162, 667–675. doi: 10.1093/toxsci/kfx287

Angermueller, C., Pärnamaa, T., Parts, L., and Stegle, O. (2016). Deep learning for computational biology. *Mol. Syst. Biol.* 12:878. doi: 10.15252/msb.20156651

Ankley, G. T., Bennett, R. S., Erickson, R. J., Hoff, D. J., Hornung, M. W., Johnson, R. D., et al. (2010). Adverse outcome pathways: a conceptual framework to support ecotoxicology research and risk assessment. *Environ. Toxicol. Chem.* 29, 730–741. doi: 10.1002/etc.34

Attene-Ramos, M. S., Miller, N., Huang, R., Michael, S., Itkin, M., Kavlock, R. J., et al. (2013). The Tox21 robotic platform for the assessment of environmental chemicals – from vision to reality. *Drug Discov. Today* 18, 716–723. doi: 10.1016/J.DRUDIS.2013.05.015

Bajusz, D., Rácz, A., and Héberger, K. (2015). Why is Tanimoto index an appropriate choice for fingerprint-based similarity calculations? *J. Cheminform.* 7:20. doi: 10.1186/s13321-015-0069-3

Baldi, P. (2012). "Autoencoders, Unsupervised Learning, and Deep architectures," in *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*, (Washington, DC).

Bemis, G. W., and Murcko, M. A. (1996). The properties of known drugs. 1. Molecular frameworks. *J. Med. Chem.* 39, 2887–2893. doi: 10.1021/jm9602928

Bergstra, J., Bardenet, R., Bengio, Y., and Kégl, B. (2011). "Algorithms for hyper-parameter optimization," in *Proceeding of the 25th Annual Conference on Advances in Neural Information Processing Systems (NIPS)*, (Granada), 2546–2554.

Bultinck, P., Langenaeker, W., Lahorte, P., De Proft, F., Geerlings, P., Waroquier, M., et al. (2002). The electronegativity equalization method I: parametrization and validation for atomic charge calculations. *J. Phys. Chem. A* 106, 7887–7894. doi: 10.1021/jp0205463

Cawley, G. C., and Talbot, N. L. C. (2010). *On Over-fitting in Model Selection and Subsequent Selection Bias in Performance Evaluation*. Available at: http://jmlr.csail.mit.edu/papers/volume11/cawley10a/cawley10a.pdf (accessed October 21, 2018)

Chandra, B., and Sharma, R. K. (2015). "Exploring autoencoders for unsupervised feature selection," in *Proceedings of the 2015 International Joint Conference on Neural Networks (IJCNN)*, (Killarney: IEEE), 1–6. doi: 10.1109/IJCNN.2015.7280391

Chollet, F. (2015). *Keras*. Available at: https://github.com/keras-team/keras (accessed October 17, 2018).

Cummins, N., Baird, A., and Schuller, B. W. (2018). Speech analysis for health: current state-of-the-art and the increasing impact of deep learning. *Methods* 151, 41–54. doi: 10.1016/j.ymeth.2018.07.007

Dana, D., Gadhiya, S., St. Surin, L., Li, D., Naaz, F., Ali, Q., et al. (2018). Deep learning in drug discovery and medicine; scratching the surface. *Molecules* 23:E2384. doi: 10.3390/molecules23092384

Davis, J., and Goadrich, M. (2006). "The relationship between precision-recall and ROC curves," in *Proceedings of the 23rd International Conference on Machine learning*, (New York, NY: ACM), 233–240.

de Jésus-Tran Karine, P., Pierre-Luc, C., Line, C., Jonathan, B., Fernand, L., and Rock, B. (2006). Comparison of crystal structures of human androgen receptor ligand-binding domain complexed with various agonists reveals molecular determinants responsible for binding affinity. *Protein Sci.* 15, 987–999. doi: 10.1110/ps.051905906

Dudek, A. Z., Arodz, T., and Gálvez, J. (2006). Computational methods in developing quantitative structure-activity relationships (QSAR): a review. *Comb. Chem. High Throughput Screen.* 9, 213–228. doi: 10.2174/138620706776055539

European Union (2006). *Regulation (EC) No 1907/2006 - Registration, Evaluation, Authorisation and Restriction of Chemicals (REACH)*. Available at: https://osha.europa.eu/en/legislation/directives/regulation-ec-no-1907-2006-of-the-european-parliament-and-of-the-council (accessed June 3, 2018).

Fernandez, M., Ban, F., Woo, G., Hsing, M., Yamazaki, T., LeBlanc, E., et al. (2018). Toxic colors: the use of deep learning for predicting toxicity of compounds merely from their graphic images. *J. Chem. Inf. Model.* 58, 1533–1543. doi: 10.1021/acs.jcim.8b00338

Gao, M., Igata, H., Takeuchi, A., Sato, K., and Ikegaya, Y. (2017). Machine learning-based prediction of adverse drug effects: an example of seizure-inducing compounds. *J. Pharmacol. Sci.* 133, 70–78. doi: 10.1016/j.jphs.2017.01.003

Geidl, S., Bouchal, T., Raček, T., Svobodová Vareková, R., Hejret, V., Krenek, A., et al. (2015). High-quality and universal empirical atomic charges for chemoinformatics applications. *J. Cheminform.* 7:59. doi: 10.1186/s13321-015-0107-1

Huang, R., and Xia, M. (2017). Editorial: Tox21 challenge to build predictive models of nuclear receptor and stress response pathways as mediated by exposure to environmental toxicants and drugs. *Front. Environ. Sci.* 5:3. doi: 10.3389/fenvs.2017.00003

Huang, R., Xia, M., Nguyen, D.-T., Zhao, T., Sakamuru, S., Zhao, J., et al. (2016a). Tox21Challenge to build predictive models of nuclear receptor and stress response pathways as mediated by exposure to environmental chemicals and drugs. *Front. Environ. Sci.* 3:85. doi: 10.3389/fenvs.2015.00085

Huang, R., Xia, M., Sakamuru, S., Zhao, J., Shahane, S. A., Attene-Ramos, M., et al. (2016b). Modelling the Tox21 10 K chemical profiles for in vivo toxicity prediction and mechanism characterization. *Nat. Commun.* 7:10425. doi: 10.1038/ncomms10425

Huang, S., Li, X., Zhang, Z., He, Z., Wu, F., Liu, W., et al. (2016c). Deep learning driven visual path prediction from a single image. *IEEE Trans. Image Process.* 25, 5892–5904. doi: 10.1109/TIP.2016.2613686

Hughes, T. B., Dang, N. L., Miller, G. P., and Swamidass, S. J. (2016). Modeling reactivity to biological macromolecules with a deep multitask network. *ACS Cent. Sci.* 2, 529–537. doi: 10.1021/acscentsci.6b00162

Hughes, T. B., Miller, G. P., and Swamidass, S. J. (2015). Modeling epoxidation of drug-like molecules with a deep machine learning network. *ACS Cent. Sci.* 1, 168–180. doi: 10.1021/acscentsci.5b00131

Hughes, T. B., and Swamidass, S. J. (2017). Deep learning to predict the formation of quinone species in drug metabolism. *Chem. Res. Toxicol.* 30, 642–656. doi: 10.1021/acs.chemrestox.6b00385

Jeni, L. A., Cohn, J. F., and De La Torre, F. (2013). "Facing imbalanced data–recommendations for the use of performance metrics," in *Proceedings of the 2013 Humaine Association Conference on Affective Computing and Intelligent Interaction*, (Geneva: IEEE), 245–251.

Kim, S., Thiessen, P. A., Cheng, T., Yu, B., and Bolton, E. E. (2018). An update on PUG-REST: RESTful interface for programmatic access to PubChem. *Nucleic Acids Res.* 46, W563–W570. doi: 10.1093/nar/gky294

Koutsoukas, A., Monaghan, K. J., Li, X., and Huan, J. (2017). Deep-learning: investigating deep neural networks hyper-parameters and comparison of performance to shallow methods for modeling bioactivity data. *J. Cheminform.* 9:42. doi: 10.1186/s13321-017-0226-y

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature* 521, 436–444. doi: 10.1038/nature14539

Lenselink, E. B., ten Dijke, N., Bongers, B., Papadatos, G., van Vlijmen, H. W. T., Kowalczyk, W., et al. (2017). Beyond the hype: deep neural networks outperform established methods using a ChEMBL bioactivity benchmark set. *J. Cheminform.* 9:45. doi: 10.1186/s13321-017-0232-0

Li, Y., Idakwo, G., Thangapandian, S., Chen, M., Hong, H., Zhang, C., et al. (2018). Target-specific toxicity knowledgebase (TsTKb): a novel toolkit for in silico predictive toxicology. *J. Environ. Sci. Heal. Part C* 36, 1–18. doi: 10.1080/10590501.2018.1537148

Liu, R., Madore, M., Glover, K. P., Feasel, M. G., and Wallqvist, A. (2018). Assessing deep and shallow learning methods for quantitative prediction of acute chemical toxicity. *Toxicol. Sci.* 164, 512–526. doi: 10.1093/toxsci/kfy111

Mayr, A., Klambauer, G., Unterthiner, T., and Hochreiter, S. (2016). DeepTox: toxicity prediction using deep learning. *Front. Environ. Sci.* 3:80. doi: 10.3389/fenvs.2015.00080

Mayr, A., Klambauer, G., Unterthiner, T., Steijaert, M., Wegner, J. K., Ceulemans, H., et al. (2018). Large-scale comparison of machine learning methods for drug target prediction on ChEMBL. *Chem. Sci.* 9, 5441–5451. doi: 10.1039/c8sc00148k

Min, S., Lee, B., and Yoon, S. (2016). Deep learning in bioinformatics. *Brief. Bioinform* 18, 851–869. doi: 10.1093/bib/bbw068

National Research Council (ed.) (2007). *Toxicity Testing in the 21st Century: A Vision and A Strategy*. Washington, DC: National Academies Press.

Névéol, A., Zweigenbaum, P., and Section Editors for the Imia Yearbook Section on Clinical Natural Language Processing (2018). Expanding the diversity of texts and applications: findings from the section on clinical natural language processing of the international medical informatics association yearbook. *Yearb. Med. Inform.* 27, 193–198. doi: 10.1055/s-0038-1667080

O'Boyle, N. M., Banck, M., James, C. A., Morley, C., Vandermeersch, T., and Hutchison, G. R. (2011). Open babel: an open chemical toolbox. *J. Cheminform.* 3:33. doi: 10.1186/1758-2946-3-33

Organization for Economic Co-operation and Development [OECD] (2013). *Guidance Document on Developing and Assessing Adverse Outcome Pathways*. Paris: OECD environment, health and safety publications.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). Scikit-learn: machine learning in python. *J. Mach. Learn. Res.* 12, 2825–2830.

Plewczynski, D., Spieser, S. A. H., and Koch, U. (2006). Assessing different classification methods for virtual screening. *J. Chem. Inf. Model.* 46, 1098–1106. doi: 10.1021/ci050519k

Powers, D. M. W. (2011). Evaluation: from precision, recall and F-Measure to roc, informedness, markedness & correlation. *J. Mach. Learn. Technol.* 2, 37–63.

Saito, T., Rehmsmeier, M., Hood, L., Franco, O., Pereira, R., and Wang, K. (2015). The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PLoS One* 10:e0118432. doi: 10.1371/journal.pone.0118432

Shao, C.-Y., Chen, S.-Z., Su, B.-H., Tseng, Y. J., Esposito, E. X., and Hopfinger, A. J. (2013). Dependence of QSAR models on the selection of trial descriptor sets: a demonstration using nanotoxicity endpoints of decorated nanotubes. *J. Chem. Inf. Model.* 53, 142–158. doi: 10.1021/ci3005308

Shen, D., Wu, G., and Suk, H.-I. (2017). Deep learning in medical image analysis. *Annu. Rev. Biomed. Eng.* 19, 221–248. doi: 10.1146/annurev-bioeng-071516-044442

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., van den Driessche, G., et al. (2016). Mastering the game of go with deep neural networks and tree search. *Nature* 529, 484–489. doi: 10.1038/nature16961

Snoek, J., Larochelle, H., and Adams, R. P. (2012). Practical bayesian optimization of machine learning algorithms. *Adv. Neural Inf. Process. Syst.* 25, 2960–2968.

Stokes, W. S. (2015). Animals and the 3Rs in toxicology research and testing: the way forward. *Hum. Exp. Toxicol.* 34, 1297–1303. doi: 10.1177/0960327115598410

Tan, M. E., Li, J., Xu, H. E., Melcher, K., and Yong, E. (2015). Androgen receptor: structure, role in prostate cancer and drug discovery. *Acta Pharmacol. Sin.* 36, 3–23. doi: 10.1038/aps.2014.18

Thangapandian, S., Shalini, J., Sugunadevi, S., and Woo, L. K. (2010). Docking-enabled pharmacophore model for histone deacetylase 8 inhibitors and its application in anti-cancer drug discovery. *J. Mol. Graph. Model.* 29, 382–395. doi: 10.1016/j.jmgm.2010.07.007

Tirado-Rives, J., and Jorgensen, W. L. (2006). Contribution of conformer focusing to the uncertainty in predicting free energies for protein-ligand binding. *J. Med. Chem.* 49, 5880–5884. doi: 10.1021/jm060763i

Trott, O., and Olson, A. J. (2010). AutoDock Vina: improving the speed and accuracy of docking with a new scoring function, efficientoptimization, and multithreading. *J. Comput. Chem.* 31, 455–461. doi: 10.1002/jcc

Wu, K., and Wei, G.-W. (2018). Quantitative toxicity prediction using topology based multitask deep neural networks. *J. Chem. Inf. Model.* 58, 520–531. doi: 10.1021/acs.jcim.7b00558

Wu, Y., and Wang, G. (2018). Machine learning based toxicity prediction: from chemical structural description to transcriptome analysis. *Int. J. Mol. Sci.* 19:2358. doi: 10.3390/ijms19082358

Xu, Y., Dai, Z., Chen, F., Gao, S., Pei, J., and Lai, L. (2015). Deep learning for drug-induced liver injury. *J. Chem. Inf. Model.* 55, 2085–2093. doi: 10.1021/acs.jcim.5b00238

Xu, Y., Pei, J., and Lai, L. (2017). Deep learning based regression and multiclass models for acute oral toxicity prediction with automatic chemical feature extraction. *J. Chem. Inf. Model.* 57, 2672–2685. doi: 10.1021/acs.jcim.7b00244

Yap, C. W. (2011). PaDEL-descriptor: an open source software to calculate molecular descriptors and fingerprints. *J. Comput. Chem.* 32, 1466–1474. doi: 10.1002/jcc.21707

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Frontiers
Avenue du Tribunal-Fédéral 34
1005 Lausanne | Switzerland

Visit us: www.frontiersin.org
Contact us: info@frontiersin.org | +41 21 510 17 00