



MildInt: Deep Learning-Based Multimodal Longitudinal Data Integration Framework

Garam Lee^{1,2}, Byungkon Kang¹, Kwangsik Nho^{3,4}, Kyung-Ah Sohn^{1*} and Dokyoon Kim^{2,5,6*}

¹ Department of Software and Computer Engineering, Ajou University, Suwon, South Korea, ² Biomedical & Translational Informatics Institute, Geisinger, Danville, PA, United States, ³ Center for Computational Biology and Bioinformatics, Indiana University School of Medicine, Indianapolis, IN, United States, ⁴ Center for Neuroimaging, Department of Radiology and Imaging Sciences, Indiana University School of Medicine, Indianapolis, IN, United States, ⁵ Department of Biostatistics, Epidemiology and Informatics, Perelman School of Medicine, University of Pennsylvania, Philadelphia, PA, United States, ⁶ Institute for Biomedical Informatics, University of Pennsylvania, Philadelphia, PA, United States

OPEN ACCESS

Edited by:

Tao Zeng,
Shanghai Institutes for Biological
Sciences (CAS), China

Reviewed by:

Min Chen,
Hunan Institute of Technology, China
Liansheng Wang,
Xiamen University, China

*Correspondence:

Dokyoon Kim
dokyoon.kim@pennmedicine.upenn.edu
Kyung-Ah Sohn
kasohn@ajou.ac.kr

Specialty section:

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

Received: 21 December 2018

Accepted: 13 June 2019

Published: 28 June 2019

Citation:

Lee G, Kang B, Nho K, Sohn K-A
and Kim D (2019) MildInt: Deep
Learning-Based Multimodal
Longitudinal Data Integration
Framework.
Front. Genet. 10:617.
doi: 10.3389/fgene.2019.00617

As large amounts of heterogeneous biomedical data become available, numerous methods for integrating such datasets have been developed to extract complementary knowledge from multiple domains of sources. Recently, a deep learning approach has shown promising results in a variety of research areas. However, applying the deep learning approach requires expertise for constructing a deep architecture that can take multimodal longitudinal data. Thus, in this paper, a deep learning-based python package for data integration is developed. The python package deep learning-based multimodal longitudinal data integration framework (MildInt) provides the preconstructed deep learning architecture for a classification task. MildInt contains two learning phases: learning feature representation from each modality of data and training a classifier for the final decision. Adopting deep architecture in the first phase leads to learning more task-relevant feature representation than a linear model. In the second phase, linear regression classifier is used for detecting and investigating biomarkers from multimodal data. Thus, by combining the linear model and the deep learning model, higher accuracy and better interpretability can be achieved. We validated the performance of our package using simulation data and real data. For the real data, as a pilot study, we used clinical and multimodal neuroimaging datasets in Alzheimer's disease to predict the disease progression. MildInt is capable of integrating multiple forms of numerical data including time series and non-time series data for extracting complementary features from the multimodal dataset.

Keywords: multimodal deep learning, data integration, gated recurrent unit, Alzheimer's disease, python package

INTRODUCTION

As the amount of biomedical datasets grows exponentially, the development of relevant data integration methods that can extract biological insight by incorporating heterogeneous data is required (Larranaga et al., 2006). Recently, deep learning approaches have shown promising results in numerous applications such as natural language processing, computer vision, and speech recognition. In addition, in the field of translational research, deep learning-based predictive models have shown comparable results (Chaudhary et al., 2017; Choi et al., 2017; Lu et al., 2018; Lee et al., 2019). In previous studies, they integrated multiple domains of data using deep

learning models to discover integrative features that cannot be explained by a single domain of data. For example, multimodal neuroimaging dataset is combined in (Lu et al., 2018) using deep learning-based framework for discriminating cognitively normal with Alzheimer's disease (AD), which resulted in considerable performance improvement. For the multi-omics data integration, RNA-seq, miRNA-seq, and methylation data from The Cancer Genome Atlas (TCGA) are incorporated using auto-encoder for predicting hepatocellular carcinoma survival (Chaudhary et al., 2018). Furthermore, in (Lee et al., 2019), multimodal gated recurrent unit (GRU) is used to integrate cognitive performance, cerebrospinal fluid (CSF), demographic data, and neuroimaging data to predict AD progression. Data integration is believed to help improve the classification performance by extracting complementary information from each domain of source.

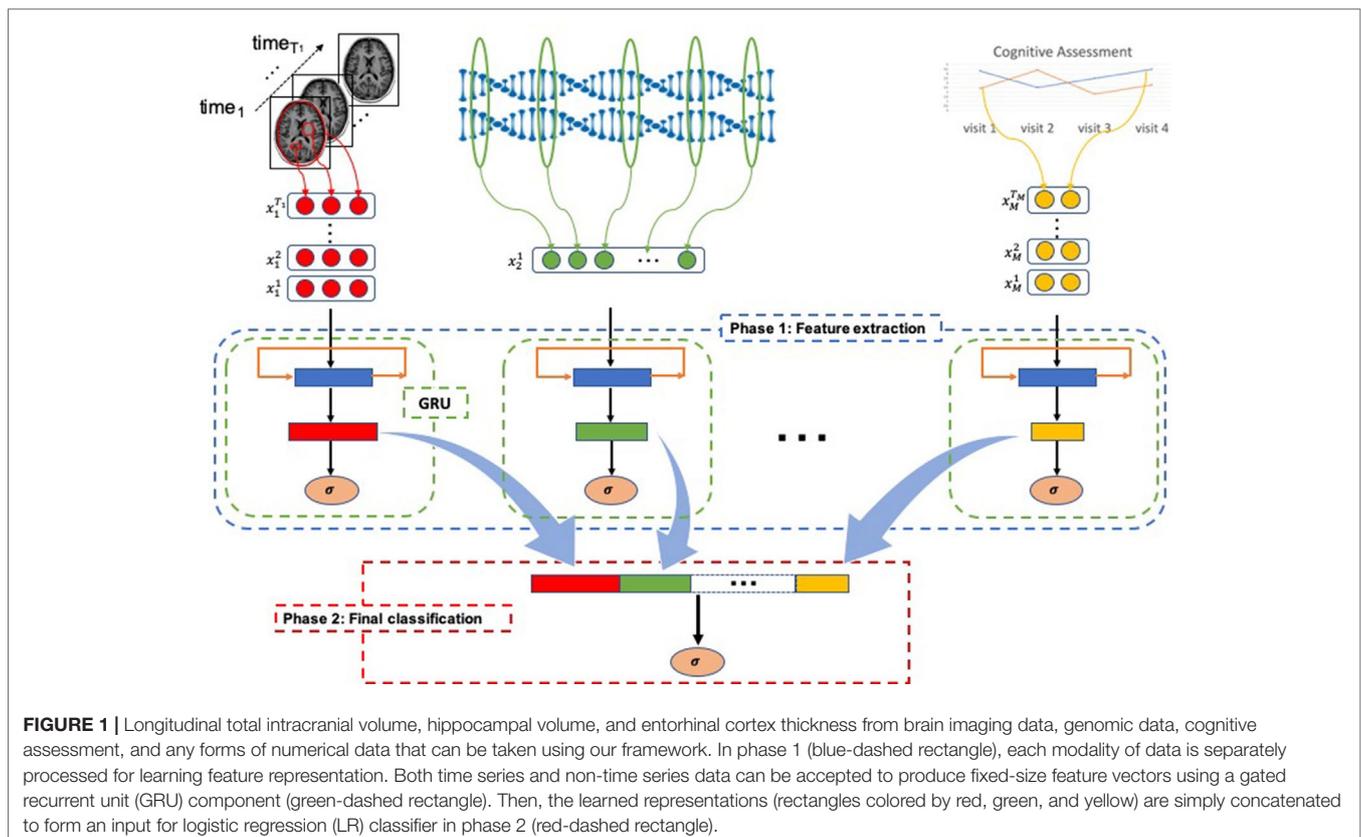
However, integrating heterogeneous data is a challenging task. First of all, multimodal data might hinder learning complementary feature representation due to the presence of mutually exclusive data, that is, a useful feature representation of the data might not be learned well since the task-irrelevant portion of the data could interfere with the task-relevant portion. In addition, dealing with datasets that consist of multiple time points is another issue for data integration. Time series data include multiple time points of data, whose length is varied over samples, while non-time series consists of a single time point of data. Thus, additional transformation steps for time series dataset

should be preceded to convert the variable-length sequence data into fixed-size representations without losing information. Finally, most commonly, the more various datasets are used, the less sample size is available. Traditional data integration methods use only samples overlapped by all modalities. Since only a few samples contain all modalities of data, it is inevitable to use a small portion of the samples, even though abundant samples are available.

In this paper, we provide a deep learning-based python package for heterogeneous data integration. The most significant advantage of our package is the flexibility in which irregular time series data are processed. As the main component of our package, we combine multiple GRUs with simple concatenation-based vector integration, which makes it possible to incorporate any number of modalities. Furthermore, nonoverlapping samples, as well as overlapping samples, can be used for training a classifier. To demonstrate the validity of our package, we conduct experiments on simulation data and real data. For simulation data, we generate multimodal time series data using the autoregressive model and solve a binary classification task. For the real data, as a pilot test, patients with mild cognitive impairment (MCI) is used to predict AD progression.

Methods

As shown in **Figure 1**, MildInt comprise two learning phases: 1) feature extraction from each modality of data and 2) learning



the integrative feature representation to make the final prediction. In phase 1, time series data from a single domain is transformed into a fixed-size vector. Then, vectors from each modality of data are integrated and fed to logistic regression (LR) classifier for the final decision making in phase 2. We use GRU as our main component for learning feature representation from the time series data. Additionally, we apply the concatenation-based data integration method to integrate multiple sources of data into a single vector.

PHASE 1: FEATURE EXTRACTION FROM EACH SINGLE MODAL TIME SERIES DATA

Recurrent Neural Network

Recurrent neural network (RNN) is a class of deep learning architecture composed of multiple recurring processing layers to learn a representation of sequential data (LeCun et al., 2015). An RNN processes an input sequence one element at a time and updates its memory state that implicitly contains information about the history of all the past elements of the sequence. The memory state is represented as a Euclidean vector (i.e., a sequence of real numbers) and is updated recursively from the input at the given step and the value of the previous memory state. Given a sequence $X = \{x_1, x_2, \dots, x_t, \dots, x_T\}$ memory state and output at each time step are computed as follows:

$$s_t = \tanh(U^{(s)}x_t + W^{(s)}s_{t-1}) \tag{1}$$

$$o_t = \text{softmax}(V^{(o)}s_t) \tag{2}$$

where U , W , and V are parameters to be learned for computing input, memory state, and output, respectively. Output is resulted from softmax function whose role is to convert the vector of hidden state into a probability vector *via* the following operation:

$$\sigma(u_i) = \frac{e^{u_i}}{\sum_k e^{u_k}} \tag{3}$$

where u_i is the i -th element of the vector u and k is the number of labels. Finally, loss function is defined with cross-entropy to quantify the distance between true label and estimated one. In our package, only the last output o_T is picked and used for the estimate because the output is regarded to carry the past features relevant to estimation.

In natural language processing, speech recognition, and anomaly detection in time series, RNN is popularly used for analyzing the sequence of words and time series data (Deng et al., 2013). One of the main advantages of using RNN is that variable length of time series data can be processed. This advantage is a critical part of our framework that is capable of accepting any variable length of time series data. However, extracting features in a long sequence of data is hard for RNN, which is known as a long-term dependency problem (Bengio et al., 1994). To handle

this problem, long short-term memory (LSTM) and GRU have been developed and practically used.

Learning Feature Representation Using Gated Recurrent Unit

GRU and LSTM are the extension of RNN in which additional parameters regulate the memory state, making it possible to “forget” irrelevant, outdated past information. Although both LSTM and GRU can handle long-term dependency problem, we selected GRU as the main component of MildInt. Since GRU has fewer parameters than LSTM, it is expected that GRU is easier for training in the field of translational informatics where only a few samples are available.

Regulating long-term information is handled by reset and update gates. Parameters for both gates are learned for determining how x_t is processed [equation (4)–(7)]. Update gate decides how amount of the previous memory value s_{t-1} is passed on. Suppose z_t is computed as 1 by equation (4), then only the previous memory is passed on, while newly computed hidden value h_t will be forgotten [equation (7)]. On the other hand, reset gate manipulates the computation between previous memory s_{t-1} and the current input x_t . In equation (6), reset gate determines the amount of previous memory value s_{t-1} . Note that GRU is a general case of RNN because setting r_t to 1 and z_t to 0 for $t = 1, 2, \dots, T$ leads GRU to functioning exactly the same as RNN.

$$z_t = \sigma(U^{(z)}x_t + s_{t-1}W^{(z)}) \tag{4}$$

$$r_t = \sigma(U^{(r)}x_t + s_{t-1}W^{(r)}) \tag{5}$$

$$h_t = \tanh(U^{(h)}x_t + (s_{t-1} \circ r_t)W^{(h)}) \tag{6}$$

$$s_t = (1 - z_t) \circ h_t + z_t \circ s_{t-1} \tag{7}$$

In **Figure 1**, x_m^t represents m -th modality of data at t time point. T_m is the maximum time length of m -th modality. A single GRU takes each modality of time series data separately for learning fixed-length representation in the first phase. Note that every modality of data is assumed to be a time series data in our package. For the single time point modalities, they are also considered as length-1 time series data for ease of integration. Without multiple time points of input data, GRU is only a fully connected network with a prior hidden state. Thus, the GRU component is able to take not only time series data but also non-time series data as well. The feature representations learned in the first phase are optimized only by a single modality of data. Thus, phase 1 can be used for a feature learning phase from a single domain of source.

PHASE 2: FINAL CLASSIFICATION

In the second phase, integration of multiple domains of data takes place. The feature representations are learned separately in the first phase. Thus, a vector produced from a GRU component

contains only the information of a single modality. For learning integrative feature representation in the second phase, vectors are simply concatenated (**Figure 1**). Based on the concatenated vector, any classification algorithm can be used in phase 2. In our package, we provide LR because it yields good interpretability by analyzing beta coefficients of the trained classifier. Also, in the experiments with real data and simulation data, an LR model was used for the final decision.

LR is a classification algorithm in which the outcome is the probability of binary classes. Sigmoid function transforms the linear combination of the input features into probability values that can be mapped to the binary class. We apply l_1 -regularized LR for the classification. A python library Sklearn (Bengio et al., 1994) is used for LR in our package.

RESULTS

To validate the performance of our package, experiments on simulation data and real data are conducted. In the experiment with simulation data, multimodal time series data are generated and tested for binary classification. The classification performance of our package is compared with other well-known methods such as logistic regression (LR), random forest (RF), and support vector machine (SVM). In the experiment with real data, four modalities of datasets, such as cognitive performance, cerebrospinal fluid (CSF), demographic data, and MRI data of patients in Alzheimer's disease, are used for MCI conversion prediction that is also set to binary classification.

CLASSIFICATION TASK ON THE SIMULATION DATA

In this section, we demonstrate the performance improvement using multimodal data and time series data. In the first experiment, only a single time point of data is used to evaluate the performance improvement of MildInt over other prominent classification algorithms such as SVM, LR, and RF. In the following experiment, the performance of using time series data is observed to evaluate the effectiveness of applying additional time points of data.

To generate time series data for binary classification, we apply the autoregressive model. First two underlying networks A_0 and A_1 are generated for the parameters in the autoregressive model. It is assumed that individual record is generated based on the underlying network in which 0-labeled data are generated from network A_0 while 1-labeled data from A_1 . The underlying network A_0 is built in which edges are randomly selected as either 0 or 1, and a network A_1 against A_0 is built with a distance d ranging from 0 to 1 in equation (8) where A_{0j} is an element of the i -th row and the j -th column in the network A_0 whose size is $n \times n$.

$$A_{1j} = |A_{0j} - d|, \quad \text{for } 1 \leq i, j \leq n \quad (8)$$

The distance d is a value for how likely two matrices A_0 and A_1 are distinguishable. For example, if $d = 1$, then A_0 and A_1 are

opposite matrices where edges in A_0 are not in A_1 while edges in A_1 are not in A_0 . On the other hand, if $d = 0$, A_0 and A_1 are exactly the same. Thus, dataset generated with higher d is easier to be separated. Second, we pick up sets of nodes from the underlying network to make subnetworks. Each subnetwork is considered as each modality of data because each modality of data is assumed to have a part of information for understanding entire networks. Finally, time series data are generated using the nonlinear autoregressive model in equation (9) where M is a subnetwork and ε is an error term with 0 mean and 0.1 variance.

$$x^t = \sigma(Mx^{t-1} + \varepsilon), \quad \varepsilon \sim \mathcal{N}(0, 0.1) \quad (9)$$

$$x^0 \sim \mathcal{U}(-1, 1)$$

We generated 1,000 samples whose length of time points is 10. Among 1,000 samples, only 500 samples contain all modalities of data, while the rest of them have only a part of all modalities. For evaluation, we ran fivefold cross-validation 10 times in which every fold has the same ratio of positive and negative samples.

In **Figure 2**, we only used a single time point of data to compare the classification performance depending on modality. **Figure 2A** shows inconsistent accuracies of SVM, RF, LR, and MildInt over distances since single modality of data does not contain enough information for understanding whole underlying networks. Thus, the performance becomes more affected by the error term. Contrary to the performance with single modality, performance using multi-modality of data is less affected by error term. As shown in **Figure 2B**, accuracy is improved consistently over distances from 0.5 to 1.0. In particular, the performance of MildInt shows 1.0 accuracy over distances from 0.8 to 1.0 since MildInt can take non-overlapping as well as overlapping samples on input, while SVM, RF, and LR can only use overlapping samples.

From **Figure 3**, we can see the effectiveness of using time series data. As increasing the number of time points, the performance using single modality is consistently increased (**Figure 3A**). Using multi-modality of time series data whose length is more than 6, two sets of data are perfectly classified from the distance 0.5 to 1.0 as seen in **Figure 3B**. Intuitively, data from multiple time points have more information than data at a single time point. Thus, MildInt can exploit temporal changes in time series data for the correct classification.

CLASSIFICATION TASK ON THE REAL DATASET

For the experiment with real data, we used 865 subjects in MCI obtained from Alzheimer's disease neuroimaging initiative cohort (ADNI) for predicting AD progression. The overall objective of ADNI is to test whether neuroimaging, biological markers, clinical, neuropsychological assessment could be combined to measure the AD progression. We downloaded four modalities of data including cognitive performance, CSF, magnetic resonance imaging (MRI), and demographic information; each of which has 802, 601, 865, and 865 samples, respectively, from the ADNI data repository (<http://adni.loni.usc.edu>). Informed consent was obtained for all subjects,

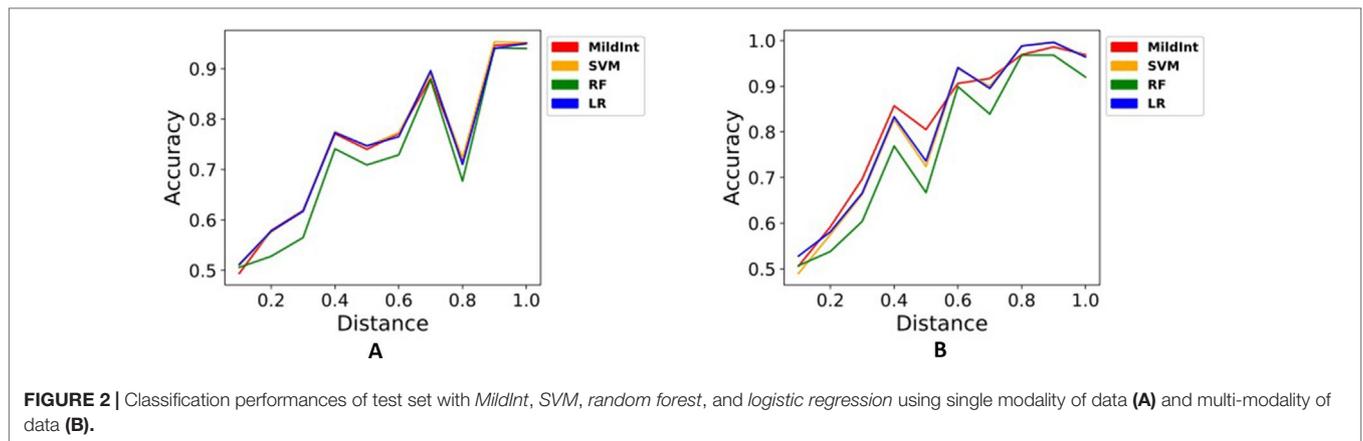


FIGURE 2 | Classification performances of test set with *MildInt*, *SVM*, *random forest*, and *logistic regression* using single modality of data (A) and multi-modality of data (B).

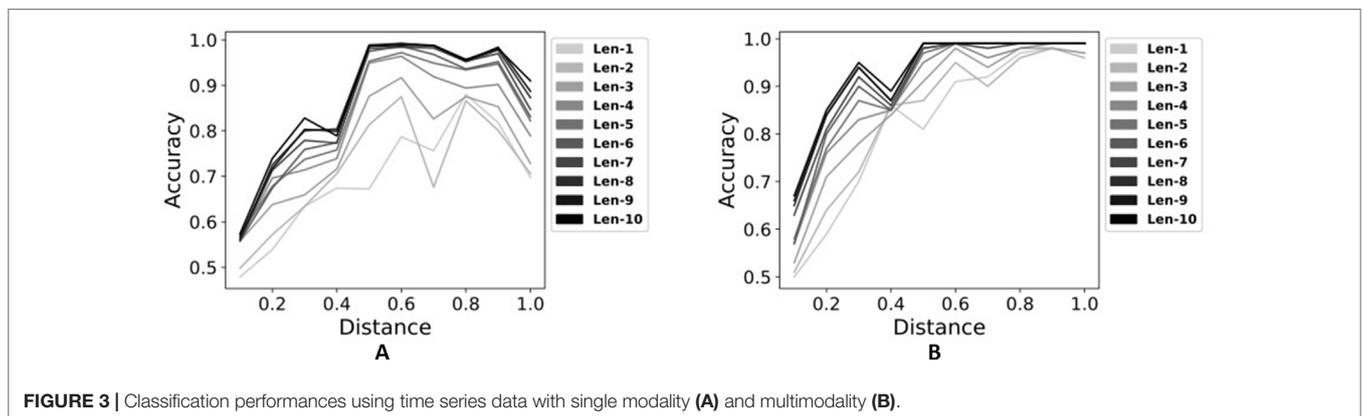


FIGURE 3 | Classification performances using time series data with single modality (A) and multimodality (B).

and the study was approved by the relevant institutional review board at each data acquisition site (for up-to-date information, see http://adni.loni.usc.edu/wp-content/themes/freshnews-dev-v2/documents/policy/ADNI_Acknowledgement_List%205-29-18.pdf). All methods were performed in accordance with the relevant guidelines and regulations. Among the four modalities of samples, 601 overlapping samples are available with 200 MCI converter and 401 MCI non-converter samples. Cognitive performance and CSF are time series data with lengths of 4.05 and 1.69 on average. MRI and demographic information are considered as length-1 time series data in our package. Note that all modalities are given in numerical vector forms. For example, we extracted gender, age, level of education, and cognitive assessment from patients' record. Especially for MRI data, a preprocessing was performed to extract features, such as total intracranial volume, hippocampal volume, and entorhinal cortex thickness, which are relevant to predicting MCI conversion. Recent methods (Lama et al., 2017; Sandeep et al., 2017) that extract features also can be used before running our package. The summary statistics of samples and hyperparameters are shown in **Table 1**.

Figure 4 shows the accuracies of our package using time series data. We removed the accuracy from the model with demographic data because the prediction performance was too low. The performance improvement using time series data is marginal due to the sparsity of time points. More than half

of the samples contain missing values, and even the length of time points is short. Furthermore, we have longitudinal samples for only two modalities of data (cognitive performance and CSF). Thus, it is hardly expected that the performance is enhanced using longitudinal data. However, classification accuracy was improved using multiple domains of data. As seen in **Figure 4**, integrating four sources of data shows the best predictive performance compared with the performance with single modalities. Finally, we compared the performance of *MildInt* with previously developed methods for MCI conversion prediction. As observed in **Table 2**, *MildInt* showed comparable prediction results.

TABLE 1 | Summary statistics for data and hyperparameters in the experiment with real data.

	#Features	Hidden dimension	Time length (avg)	Time length (sd)
Cognitive performance	2	3	4.05	1.71
Demographic information	4	5	1	0
CSF	5	6	1.69	0.95
MRI	3	4	1	0

CSF, cerebrospinal fluid; MRI, magnetic resonance imaging.

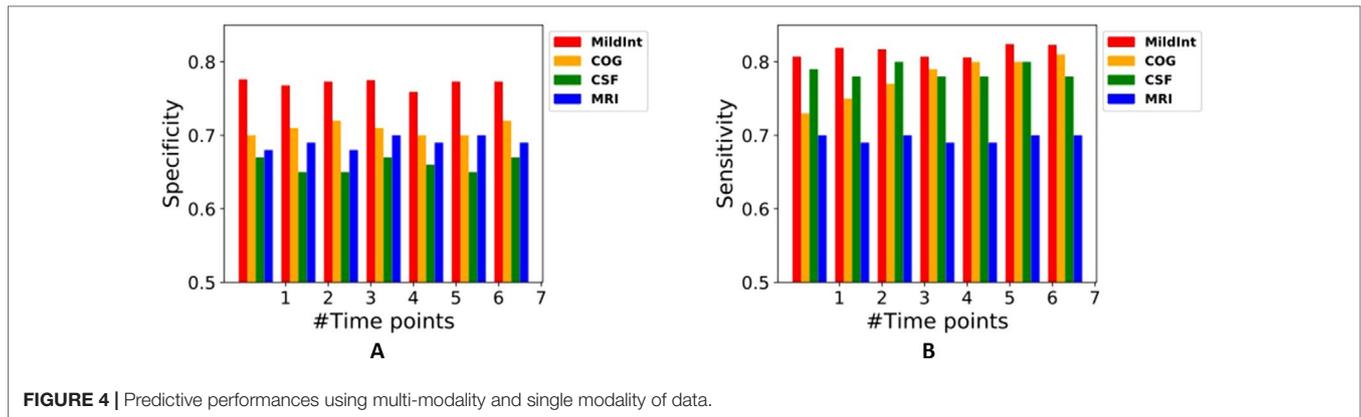


TABLE 2 | A list of previous models that train classifiers mainly with mild cognitive impairment (MCI) samples.

Method	Subjects (MCI-C/MCI-NC)	Data source	ACC	SEN	SPE
SVM (Zhang and Shen, 2012a)	43/48	MRI, PET, CSF	0.73	0.68	0.73
SVM (Cheng et al., 2012)	43/56	MRI, FDG-PET, CSF	0.79	0.84	0.72
SVM (Zhang and Shen, 2012b)	35/50	MRI, PET, cognitive score	0.78	0.79	0.78
Gaussian process (Young et al., 2013)	47/96	MRI, PET, CSF, APOE genotype	0.68	0.90	0.52
Hierarchical ensemble (Huang et al., 2017)	70/61	MRI	0.79	0.86	0.78
Deep neural network (Lu et al., 2018)	235/409	MRI, PET	0.82	0.79	0.83
MildInt	163/376	Cognitive score, neuroimaging data, CSF biomarker, demographic data	0.79	0.83	0.77

MCI-C, MCI-Converter; MCI-NC, MCI-NonConverter; ACC, Accuracy; SEN, Sensitivity; SPE, Specificity; APOE, Apolipoprotein E; FDG, Fluorodeoxyglucose.

CONCLUSION

MildInt provides multimodal GRU for heterogeneous data integration. The main advantage of our framework is that variable-length time series data and multimodal data can be processed. In addition, every available sample from all modalities including non-overlapping samples can be used for training classifier. The performance of MildInt is evaluated with simulation data and real data. In the experiment with simulation data, it showed the best performance when multimodal data and time series data were integrated. Additionally, in the experiment with real data, integrating cognitive performance, demographic information, CSF, and MRI imaging data show the best performance for MCI conversion prediction. Also, any numerical form of data such as gene expression, methylation, and single nucleotide polymorphism data can be combined in our package. MildInt is suitable to use in cases where time series data such as multiple time points of methylation data and non-time series data such as single nucleotide polymorphism should be incorporated for learning integrative feature representation. Furthermore, compared with previously developed methods, MildInt showed comparable prediction ability that can efficiently incorporate multiple domains of resources.

REQUIREMENTS

This package works on python 2.7.x in platforms such as Mac OS X, Windows, and Linux. MildInt requires python packages

such as Pandas, Numpy, Tensorflow, and Sklearn to be installed independently. To make MildInt fully functioning, Tensorflow with graphics processing units (GPU) from NVIDIA should be equipped. The GPU-enabled version of Tensorflow has requirements such as 64-bit Linux, NVIDIA CUDA 7.5 (CUDA 8.0 required for Pascal GPUs), and NVIDIA, cuDNN v4.0 (minimum) or v5.1 (recommended).

AUTHOR CONTRIBUTIONS

This study was conceived by GL, K-AS, and DK. Experiments were designed and performed by all authors. The manuscript was initially written by GL. All the authors revised the manuscript and approved the final version prior to submission.

FUNDING

The support for this research was provided by NLM R01 LM012535, NIA R03 AG054936, and the Pennsylvania Department of Health (#SAP 4100070267). The department specifically disclaims responsibility for any analyses, interpretations, or conclusions. This work was also supported by the National Research Foundation of Korea grant funded by the Korea government (MSIT) (no. NRF-2019R1A2C1006608).

REFERENCES

- Bengio, Y., Simard, P., and Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Trans. Neural Networks* 5 (2), 157–166. doi: 10.1109/72.279181
- Chaudhary, K., Poirion, O. B., Lu, L., and Garmire, L. X. (2017). Deep learning based multi-omics integration robustly predicts survival in liver cancer. *Clin. Can. Res.* 0853, 1246–1259. doi: 10.1101/114892
- 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. Can. Res.* 24 (6), 1248–1259. doi: 10.1158/1078-0432.CCR-17-0853
- Cheng, B., Liu, M., Zhang, D., Musell, B.C., Shen D. (2015). Domain Transfer Learning for MCI Conversion Prediction. *IEEE Trans. Biomed. Eng.* 62 (7), 1805–1817. doi: 10.1109/TBME.2015.2404809
- Choi, E., Schuetz, A., Stewart, W. F., and Sun, J. (2017). Using recurrent neural network models for early detection of heart failure onset. *J. Am. Med. Inf. Assoc.* 24 (2), 361–370.
- Deng, L., Hinton, G., and Kingsbury, B. (2013). New types of deep neural network learning for speech recognition and related applications: An overview. In: *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on: 2013*. IEEE: 8599–8603. doi: 10.1109/ICASSP.2013.6639344
- Huang, M., Yang, W., Feng, Q., Chen, W., Weiner, M. W., Aisen, P., et al. (2017). Longitudinal measurement and hierarchical classification framework for the prediction of Alzheimer's disease. *Sci. Rep.* 7, 39880. doi: 10.1038/srep39880
- Lama, R. K., Gwak, J., Park, J.-S., and Lee, S.-W. (2017). Diagnosis of Alzheimer's disease based on structural MRI images using a regularized extreme learning machine and PCA features. *J. Healthcare Eng.* 2017, 11. doi: 10.1155/2017/5485080
- Larranaga, P., Calvo, B., Santana, R., Bielza, C., Galdiano, J., Inza, I., et al. (2006). Machine learning in bioinformatics. *Briefings Bioinf.* 7 (1), 86–112. doi: 10.1093/bib/bbk007
- LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature* 521 (7553), 436. doi: 10.1038/nature14539
- Lee, G., Nho, K., Kang, B., Sohn, K.A., Kim, D. (2019). Predicting Alzheimer's disease progression using multi-modal deep learning approach. *Sci. Rep.* 9(1), 1952. doi: 10.1038/s41598-018-37769-z
- Lu, D., Popuri, K., Ding, G. W., Balachandar, R., and Beg, M. F. (2018). Multimodal and multiscale deep neural networks for the early diagnosis of Alzheimer's disease using structural mr and fdg-pet images. *Sci. Rep.* 8 (1), 5697. doi: 10.1038/s41598-018-22871-z
- Sandeep, C., Kumar, A., Mahadevan, K., and Manoj, P. (2017). Feature extraction of MRI brain images for the early detection of alzheimer's disease. *Bioprocess Eng.* 1 (2), 35–42. doi: 10.1109/I2C2.2017.8321780
- Young, J., Modat, M., Cardoso, M. J., Mendelson, A., Cash, D., and Ourselin, S. (2013). Initiative AsDN: accurate multimodal probabilistic prediction of conversion to Alzheimer's disease in patients with mild cognitive impairment. *NeuroImage Clin.* 2, 735–745. doi: 10.1016/j.nicl.2013.05.004
- Zhang, D., and Shen, D. (2012a). Initiative AsDN: multi-modal multi-task learning for joint prediction of multiple regression and classification variables in Alzheimer's disease. *NeuroImage* 59 (2), 895–907. doi: 10.1016/j.neuroimage.2011.09.069
- Zhang, D., and Shen, D. (2012b). Initiative AsDN: predicting future clinical changes of MCI patients using longitudinal and multimodal biomarkers. *PloS One* 7 (3), e33182. doi: 10.1371/journal.pone.0033182

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.

Copyright © 2019 Lee, Kang, Nho, Sohn and Kim. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.