frontiers | Frontiers in Signal Processing

TYPE Brief Research Report
PUBLISHED 21 September 2022
DOI 10.3389/frsip.2022.999457

OPEN ACCESS

EDITED BY
Nikolaos Mitianoudis,
Democritus University of Thrace,
Greece

REVIEWED BY
Francesco Beritelli,
University of Catania, Italy
Jan Holub,
Czech Technical University in Prague,
Czechia
Peter Pocta,
University of Žilina, Slovakia

*CORRESPONDENCE
Elhard Kumalija,
lijaudsm@gmail.com

SPECIALTY SECTION
This article was submitted to Audio and Acoustic Signal Processing,
a section of the journal
Frontiers in Signal Processing

RECEIVED 21 July 2022
ACCEPTED 16 August 2022
PUBLISHED 21 September 2022

CITATION
Kumalija E and Nakamoto Y (2022),
Performance evaluation of automatic
speech recognition systems on
integrated noise-network
distorted speech.
Front. Sig. Proc. 2:999457.
doi: 10.3389/frsip.2022.999457

Performance evaluation of automatic speech recognition systems on integrated noise-network distorted speech

Elhard Kumalija* and Yukikazu Nakamoto

Graduate School of Applied Informatics, University of Hyogo, Kobe, Hyogo, Japan

In VoIP applications, such as Interactive Voice Response and VoIP-phone conversation transcription, speech signals are degraded not only by environmental noise but also by transmission network quality, and distortions induced by encoding and decoding algorithms. Therefore, there is a need for automatic speech recognition (ASR) systems to handle integrated noise-network distorted speech. In this study, we present a comparative analysis of a speech-to-text system trained on clean speech against one trained on integrated noise-network distorted speech. Training an ASR model on noise-network distorted speech dataset improves its robustness. Although the performance of an ASR model trained on clean speech depends on noise type, this is not the case when noise is further distorted by network transmission. The model trained on noise-network distorted speech exhibited a 60% improvement rate in the word error rate (WER), word match rate (MER), and word information lost (WIL) over the model trained on clean speech. Furthermore, the ASR model trained with noise-network distorted speech could tolerate a jitter of less than 20% and a packet loss of less than 15%, without a decrease in performance. However, WER, MER, and WIL increased in proportion to the jitter and packet loss as they exceeded 20% and 15%, respectively. Additionally, the model trained on noise-network distorted speech exhibited higher robustness compared to that trained on clean speech. The ASR model trained on noise-network distorted speech can also tolerate signal-to-noise (SNR) values of 5 *dB* and above, without the loss of performance, independent of noise type.

KEYWORDS

audio signal processing, automatic speech recognition, deep learning, speech-to-text, voice over IP

1 Introduction

ASR systems provide services such as voice search and automatic call transcription. Thus, ASR systems has been widely implemented in health care systems, virtual assistants on mobile devices, and cognitive bots. Because speech is the most preferred and natural mode of communication between humans, the industries utilizing ASR applications will continue to expand.

Today, Voice over Internet Protocol (VoIP) is the most applied method for voice communication transmission. VoIP is a common component of applications such as social networking services, and multimedia streaming applications. To improve the user experience and quality of service, the ASR is built into VoIP applications. Few examples include AI-powered meeting transcription and transcription in call centers. The VoIP-transmitted speech presents a new challenge to the ASR systems as it originates from diverse sources and is captured in varying levels of environmental noise. This encompasses hand-free devices on cars and VoIP calls in noisy environments such as train stations and airports among other examples. Furthermore, VoIP speech signals are distorted not only by the environmental noise but also the transmission network. These characteristics of VoIP speech signals cause hindrance to the designing of robust and high accurate ASR systems.

Deep learning has outperformed other ASR techniques Li and Sim (2014). To build highly accurate and robust deep-learning-based ASR, various techniques has been studied, including feature extraction, language models, deep learning architectures, and rich characteristics datasets Malik et al. (2021). Deep learning is reliant on the availability of massive amounts of data. Thus, speech datasets for the development and evaluation of ASR systems have evolved over time.

Traditional deep-learning-based ASR systems were trained on studio-recorded read speech signal datasets. The earlier such datasets are the ATR Japanese speech database Kurematsu et al. (1990), TIMIT Acoustic-Phonetic Continuous Speech Corpus Garofolo et al. (1993), WSJ corpus Charniak et al. (2000), and Mandarin Chinese broadcasting news Wang et al. (2005). However, a speech read from a pre-prepared script lacks the naturality of everyday human speech. Furui et al. introduced a Corpus for Spontaneous Japanese Furui et al. (2000). The spontaneous monologue bears close resemblance to the natural human conversation. With the prevalence of deep learning, much larger datasets were introduced to tap into the potential of deep learning in ASR.

Deep learning based ASR models trained on large datasets tend to yield high performance. Large datasets such as LibriSpeech Panayotov et al. (2015), TED-LIUM Corpus Rousseau et al. (2012), Köhn et al. (2016) and Common Voice Ardila et al. (2020) contains a 1000s hours of speech. These large datasets have improved the performance of ASR systems. However, the performance still degrades in real application environments, where speech signals are usually captured with environmental noise.

The ASR systems trained on large datasets of studio-recorded speech exhibit low performance on noisy or degraded speech. This has led to the introduction of speech datasets recorded in natural environments such as the domestic setting, for example the DIRHA-English corpus Ravanelli et al. (2015), CHiME-2 Barker et al. (2013), CHiME-3 Barker et al. (2015), CHiME-5 and Barker et al. (2018). Natural environment speech datasets encompasses recorded speech spoken live in noisy environments and simulated speech datasets that were generated by artificially mixing the clean speech data with noisy backgrounds. The introduction of speech datasets recorded in natural, noisy environments has improved the robustness of ASR. However, the speech may also get distorted as a result of degradation that occurs when it is transmitted through an IP network.

Although Environmental noise degrades the speech quality, distortion and degradation are also introduced in the transmission of speech through computer networks. Low bandwidth, echo, encoding–decoding distortion, differences in handsets, and network poor quality all present new challenges toward building robust and highly accurate ASR systems. For the CTIMIT Brown and George (1995) dataset generated by transmitting clean voice speech through a cellular network, network distortions caused a 58% drop in the ASR performance. Training on network-distorted speech increased the recognition accuracy by 82%.

VoIP applications use packet-switched networks, which have different characteristics from that of circuit-switched networks. In the VoIP applications, the speech quality is degraded by delay, jitter, packet loss, packet burst loss, network bandwidth, encoding and decoding algorithms da Silva et al. (2008). The effect of combined noise, network and encoding parameters on deep-learning-based ASR models has not been extensively studied.

The first contribution of this paper is the analysis of the impact of noise-network speech distortions on the ASR system accuracy. This analysis is important for designing and planning VoIP-based ASR systems, such as Interactive Voice Response, as well as VoIP-phone conversations transcription. The second contribution of this paper is a discussion of the potential performance optimization of the existing ASR models pre-trained on clean speech datasets by re-training the models using integrated noise-network distorted speech. Using transfer learning, the performance of the existing ASR models can be optimized to robustly handle noise-network distorted speech.

The reminder of this paper is structured as follows. Section 2 explains the noise-network dataset characteristics and parameters, and the network emulation process used to generate the noise-network distorted speech dataset. The process encompasses the generation, encoding, transmission

TABLE 1 Noise-network distortion parameters.

Distortion	Parameter	Values
Network	Packet Loss (%)	0, 10, 15, 20, 25, 30, 35
	Delay (ms)	0, 100, 200, 300, 500
	Jitter (% delay)	0, 10, 20, 30, 40
	Codec	G722
Noise	Noise type	Babble, Car, Exhibition Hall, Restaurant, Street, Airport, Train Station, Train
	SNR (dB)	0, 5, 10, 15

through a simulated network environment, and decoding of noisy speech, in order to generate the noise-network distorted speech. Section 3 presents the experiment setup for a performance evaluation of the ASR on noise-network distorted speech dataset. We evaluated two ASR speech models: the ASR model trained on clean speech dataset (CSM) and the ASR model trained on noise-network speech dataset (NNSM). The NNSM was trained by fine-tuning a pre-trained CSM on noise-network distorted speech. The evaluation examines the effects of noise type and network conditions such as delay, jitter and packet loss on the performance of the ASR systems. The performance of the two models is compared on clean speech dataset, and noise-network distorted speech dataset. Section 4 presents the evaluation results. Finally, section 5 concludes our contribution and suggests future directions of research.

2 Dataset

The datasets used for the training and testing of deep-learning-based ASR systems has evolved from clean-read speech, spontaneous-speech, large dataset size speech corpus, artificially added environmental noise, speech recorded in domestic environments, and speech transmitted through cellular networks. The proposed dataset aims to build on the existing datasets with the addition of distortions induced by network conditions and encoding-decoding schemes on speech transmitted through the VoIP system. This dataset is built on clean speech, which is then distorted by noise at different signal-to-noise ratios (SNR), then transmitted at different network quality of service (QoS) parameters to generate noise-network distorted speech dataset.

2.1 Acoustic characteristics

Noisy speech signals were obtained from a noisy speech corpus (NOIZEUS) Hu and Loizou (2007). This database contains 30 IEEE sentences produced by three male and three female speakers, recorded in a sound-proof booth, and then artificially corrupted by eight different real-world noises *see* Table 1. The NOIZEUS database was selected, as it includes all the phonemes in the American English language.

2.2 VoIP network characteristics

VoIP network QoS characteristics include delay, jitter, bit rate, loss rate, and loss distribution. VoIP application parameters that affect speech quality can also include Encoding–decoding parameters, such as bit rate and Forward error correction. The impact of these factors on the perceived QoS in VoIP communications has been studied extensively Sun and Ifeachor (2002, 2006); da Silva et al. (2008); Hu et al. (2020). There are many network QoS parameters that can affect the transmitted speech signals quality. However, we considered only those with a high impact on QoS of VoIP applications, which are: loss rate, burst packet loss, delay, and jitter. We studied G.722 ITU-T (2005) a wideband speech codec.

Combining the VoIP and acoustic characteristics, the new noise-network distorted speech dataset characteristics are summarized in Table 1.

The dataset also included clean speech audio files, these were studio-recorded utterances with no artificially added noise. Clean speech utterances were also distorted when they were sent through the networks. The parameters were selected to closely match those of the real-world environmental noise and the internet QoS. The noise types were babble, car, exhibition hall, restaurant, street, airport, train station, and train. The network parameters were selected to encompass the characteristics of both good and poor internet quality environments.

2.3 Noise-network distorted dataset

To generate the noise-network distorted speech database, the clean speech artificially corrupted by noise was transmitted through an emulated network. We used Netem Linux

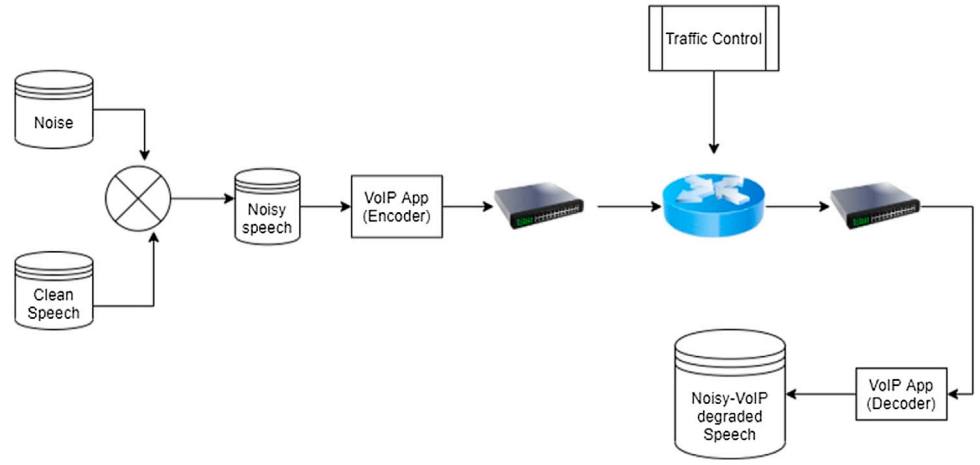

FIGURE 1
Noise-network speech dataset generation system.

Foundation (2021) network emulation software in collaboration with Tc Hubert (2001), a tool used to configure traffic control in the Linux kernel. Netem and Tc provide network emulation functionalities to emulate the properties of wide area networks. The Netem is a kernel component which can be enabled or disabled. In recent Linux distributions, Netem is already enabled and Tc software is pre-installed.

A speech generation environment was set up as shown in Figure 1. The router had the following hardware specifications: SoC Broadcom BCM2837 1.2 GHz ARM Cortex-A53 Quad Core Processor (ARMv8 Family), Memory: 1 GB LPDDR2 running Debian operating system (OS). The Debian OS had Netem and Tc enable Linux kernel. FFmpeg version 4.2.4 The FFmpeg developers (2020) the open-source command-line tool for converting multimedia formats was used to encode and decode the speech signals. Encoding and decoding application platform was Intel(R) Core(TM) i5-10210U CPU @ 1.60 GHz, 23 GB RAM, 2 TB drive, running Ubuntu OS version 20.04. Speech signals were transmitted using the User Datagram Protocol.

In wide area networks, parameters such as packet loss, jitter, and delay are random variables. Several mathematical models are used to represent this randomness. For simplicity, we used a normal distribution to generate the following noise-network distorted speech delay function:

X is normally distributed with mean μ and standard deviation σ:

$$X \sim \mathcal{N}(\mu, \sigma^2) \tag{1}$$

where X is the delay distribution, μ is the mean delay, and σ is the jitter, expressed as the percentage of delay as shown in Table 1. In the case of packet loss, the loss was normally distributed with the mean $\sigma = 0$.

A total of 246,500 sentences with different acoustic and network distortions were generated from 30 utterances. Each sentence length was 2 s as a result the generated noise-network distortion dataset contained 192.5 h of speech. The generated noise-network distorted speech database was used to study the performance of the ASR systems for VoIP-based applications.

3 Performance evaluation of ASR on noise-network distorted dataset

Different ASR engines have been proposed. To evaluate the performance of ASR systems on the noise-network distorted database, we used DeepSpeech version 0.9.3 Mozilla (2020). DeepSpeech is an open-source speech-to-text engine that uses a model trained by machine-learning techniques based on Baidu's study Hannun et al. (2014). This technique does not require hand-designed features to model background noise, reverberation, or phoneme dictionary. Instead, it relies on large amounts of varied data for training. The aforementioned features render DeepSpeech Engine the best candidate for evaluating the performance on noise-network distorted speech. Evaluation was carried on both models, the Deepspeech CSM and NNSM.

3.1 Dataset

The noise-network distorted speech dataset was divided into a training dataset and a testing dataset. The testing data was selected from the total sample of noise-network distorted dataset using stratified random sampling. The total sample was divided

into groups of sentences with similar attributes, as shown in Table 1. In each similar attribute group, there were 30 sentences, which were further divided into utterances of male and female speakers. Then, three samples were selected from each group. Finally, through the stratified random sampling, 20% of the total sample was set for testing, while the remaining 80% was used for training. This testing dataset sampling method was selected to ensure that different sample attributes were equally represented in training and testing datasets. One hundred and fifty four hours of speech were used for training and 38.5 h of speech were used for testing.

3.2 ASR pre-trained model and fine-tuning process

The pre-trained model of DeepSpeech was trained on Fisher, LibriSpeech, Switchboard, Common Voice English, and WAMU radio-shows databases. The acoustic models were trained on American English with synthetic noise augmentation, and the model achieved a 7.06% word error rate on the LibriSpeech clean test corpus. The performance of the pre-trained DeepSpeech CSM on noise-network distorted speech dataset was analyzed, and then the model was optimized through transfer learning.

Transfer learning transfers the knowledge gained when solving one problem and applies it to a different problem in a related domain. Fine-tuning is a transfer learning technique that starts with a pre-trained model on the source task and trains it further on the target task. Fine-tuning is a common technique in computer vision tasks Kornblith et al. (2019). In ASR, fine-tuning was successfully applied in low resource languages, where models trained to recognize speech in rich resource languages were then transferred to low resource languages [Huang et al. (2013); Kermanshahi et al. (2021); Shi et al. (2018)]. To the best of our knowledge, this is the first case the transfer of knowledge gained by a model in speech–to–text conversion of clean speech is applied on speech–to–text conversion of noise-network distorted speech.

The noise-network dataset contained the same alphabet set as the dataset used to train the DeepSpeech model. Therefore, the released DeepSpeech model output layer matches noise-network data, and there were no need for a different classifier in our experiment. We fine-tuned the entire model graph with the noise-network dataset without adding new layers. In this experiment, the model parameters and architecture were equal to those of the released DeepSpeech model with the training dataset as the only difference. Hence, transfer learning on noise-network dataset was evaluated as the sole factor to the ASR performance improvement.

The training system environment had the following hardware and software specifications. Hardware specifications were: Processor: Intel® Core™i7-9750H CPU @ 2.60 GHz × 12, Graphics: NVIDIA Corporation TU117M [GeForce GTX, 1650 Mobile/Max-Q]/GeForce GTX 1650/PCIe/SSE2, Memory: 31.2 GB, Disk capacity: 1.3 TB. While the OS platform used was Ubuntu 20.04.2 LTS, 64-bit, and GNOME Version:3.36.8 with Windowing System: X11.

The open-source TensorFlow framework was used to build the model and train the network. The model network architecture was the same as that of DeepSpeech. The network was trained in six stages of 10 epochs each, and a generalization evaluation was performed at each stage. The training, testing, and validation used the following parameters: The training batch size, test batch size, and validation batch size were 112, as the dataset was in the multiples of 112. This is different from the original DeepSpeech model, which used a batch size of 128 for training, testing, and validation. As in the original model, a training learning rate of 0.0001 and dropout rate of 0.4 were used. For each stage, the generalization performance of the network was monitored using a subset of the Mozilla Common Voice Corpus 1 English dataset. This speech dataset is referred to as the clean speech in the experiment result presentation.

3.3 ASR performance metrics

An automatic ASR performance measurement is necessary for the rapid system development and the performance comparison of different ASR systems. Researchers generally report the performance of ASR systems using the Word Error Rate (WER) metric. WER is defined as the ratio of the total number of errors (substitution, deletion, and omissions) in the transcription output to the number of words in the speech signal input to the ASR system, given by the equation below.

$$WER = \frac{S + D + I}{N} \tag{2}$$

where S is the number of erroneous word substitutions, D is the number of word deletions, I is the number of insertions of false words in the ASR output, and N is the number of words actually spoken in speech input to the ASR system.

The WER does not reflect human judgment, such as the relative importance of certain words for the meaning of the message. Therefore, more intuitively appealing measures for ASR–the match error rate (MER) and word information lost metric (WIL) – were also used Morris et al. (2004). MER is the probability of a given match being incorrect, obtained by simply dividing the WER by its maximum possible value. Let H, S, D, and I denote the total number of word hits, substitutions, deletions, and insertions, respectively.

$$MER = \frac{S + D + I}{H + S + D + I} \tag{3}$$

The WIL metric is the difference between 100% word preservation and percentage on output words preserved. Where $H > S + D + I$, the word information preserved (WIP) is given by:

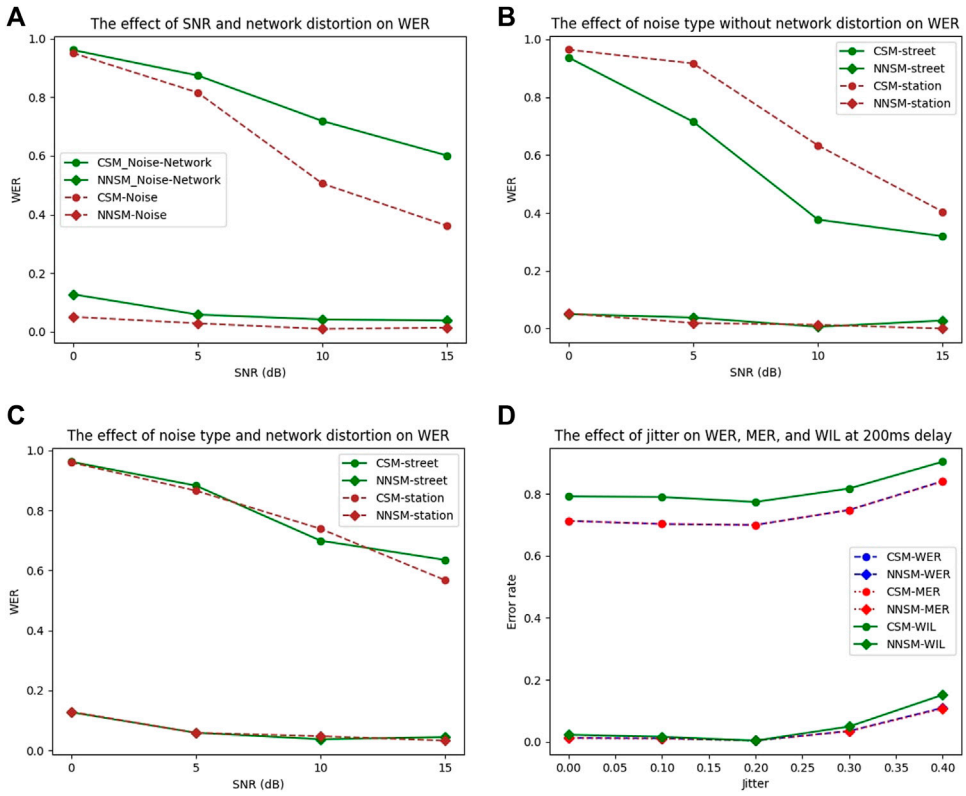

FIGURE 2
Effects of SNR, noise type, and network distortions on WER, and the effect of jitter on WER, MER and WIL. **(A)** Effects of SNR and network distortion on WER. **(B)** Effect of street noise and station noise without network distortion on WER. **(C)** Effects of street noise and station noise with network distortion on WER. **(D)** Effects of jitter on WER, MER, and WIL, for a delay of 200 ms with packet loss of 0%.

$$WIP = \frac{H}{N} \tag{4}$$

Then, WIL is derived from WIP, which is given by the equation below.

$$WIL = 1 - WIP \tag{5}$$

The DeepSpeech model performance was tested using both clean speech and noise-network speech, before and after fine-tuning with the noise-network speech dataset. The metrics used for comparison were the WER, WIL, and MER. In the next section, the results of the comparison between the performances of the two models are presented.

4 Experiment results

Fine-tuning a CSM on a noise-network distorted speech dataset improved model performance on noise-network distorted speech. However, the ASR model fine-tuned on noise-network speech undergoes a slight degradation on the generalization performance on clean speech compared to the ASR model trained on clean speech.

The WER of the DeepSpeech model on the clean speech dataset was 0.12 and 0.24 before and after fine-tuning, respectively. However, the model performance on the noise-network distorted speech dataset improved significantly from 0.79 before fine-tuning to 0.07 after fine-tuning. The ASR performance on the noise-network distorted speech improved at the expense of generalization performance, but the degradation was less when compared to the improved robustness.

The performance of CSM on clean and noise-network-distorted speech datasets was compared with that of the NNSM on the noise-network-distorted speech dataset. Generally, noise-network distortions resulted in equal degradation on WER, MER, and WIL, with WER and MER increasing from 0.19 to 0.79 and the WIL rate increasing from 0.24 to 0.85. The fine-tuning improved the model performance on WER and MER from 0.79 to 0.07, while WIL decreased from 0.85 to 0.09.

4.1 Isolated effect of noise and network distortion on WER

To examine the individual effect of noise distortion and network distortion on noisy speech data, the noise distorted speech data without any network distortion was used. Then, the general network distortion effect for each SNR was observed. The performance of the two models, the CSM and NNSM are shown in Figure 2A.

As expected, WER decreased with an increase in the SNR. However, it was noticed that the network distortion effects were high on high SNRs and did not cause significant differences to speech signals with low SNR, which were already highly distorted by noise. The fine-tuned model exhibited the same performance for SNR greater than 5 *dB*. However, the performance decreased significantly for the SNR less than 5 *dB*. Therefore, NNSM has improved robustness for speech signals with SNR greater than 5 *dB*, independent of network distortions.

4.2 Effect of noise type and network distortion on WER

Different noise types have different characteristics. We intend to understand the influence of various noise types at different SNR values on the performance of CSM, and NNSM. Moreover, we intend to understand the manner in which the network distortion impacts different noise types. Noise type affects WER, WIL, and MER differently, as shown in Figure 2B–by a comparison of street noise and train station noise. Train station noise constitutes the noise from different sources, such as approaching trains, public addressing speakers, and nearby conversations. Street noise constitutes the noise from passing cars, singing birds, and other sources. The CSM performance on station noise was lower than that of the CSM on the street noise for all SNR values. Moreover, the robustness of the fine-tuned network using noise-network distorted speech is evident, as the performance of NNSM is not affected by noise type.

When the noise-distorted speech was further distorted by network transmission errors, there was no difference in the performance of CSM and NNSM on different noise types as shown in Figure 2C. The network distortion on noisy speech masks the noise effects on WER. The WER increased with a decrease in the SNR of the speech for the CSM, but for the NNSM, the performance was the same for SNR values greater than 5. The NNSM performance on noisy speech and noise-network distorted speech data deteriorates for the SNR values of less than 5 *dB*. The NNSM can learn the effect of noise-network distortions when the SNR is greater than 5 *dB*. Hence, the NNSM is more robust than the CSM.

4.3 Effect of jitter on WER, MER, and WIL

Figure 2D shows the effect of jitter on WER, MER, and WIL. If all network parameters are constant and the jitter is less than 0.2 of delay, there is a constant effect on WER, MER, and WIL. However, with a jitter greater than 0.2 of delay, WER, MER, and WIL begin to increase proportionally to the jitter.

For the CSM, WIL is greater than MER and WER. However, for the NNSM, WER, MER, and WIL are equal when the jitter is less than 0.2 of delay, with the WIL higher than the WER and MER when the jitter is greater than 0.2 of delay.

4.4 Effect of packet loss on WER, MER, and WIL

With the jitter, delay, and burst-loss kept constant, the effect of packet loss on the clean-speech-trained model is constant for loss less than 10%. However, for a packet loss greater than 10%, the WER, MER, and WIL increased proportionally to the packet loss. By contrast, for the noise-network-trained model, the effect of an increasing packet loss begins to be seen when the loss is greater than 15%. When the packet loss is greater than 15%, the WIL error rate increase is greater than that of WER and MER, as shown in Figure 3A.

4.5 Combined effect of SNR and packet loss on WER

The combined effect of SNR and packet loss shows that both SNR and packet loss contribute significantly to the decreased performance of the clean-speech-trained ASR as shown in Figure 3B. The WER of clean-speech-trained ASR model increases with an increase in packet loss for all SNRs, whereas the WER increases with the decrease of SNR. However, for an SNR of 0 *dB*, the effect on WER is dominated by SNR rather than packet loss.

The NNSM yields significantly improved performance compared to that of the CSM. Furthermore, the NNSM shows better robustness compared to the CSM. A change in packet loss and SNR generates a small change in accuracy of the NNSM compared to that of the CSM. The NNSM performance can withstand the packet loss of less than 15% and the SNR values greater than 5 *dB* without loss of accuracy.

4.6 Combined effect of SNR and jitter on WER

An examination of the effect of jitter and SNR on ASR performance shows that the effect of SNR was significant compared to that of jitter, as shown on Figure 3C. However,

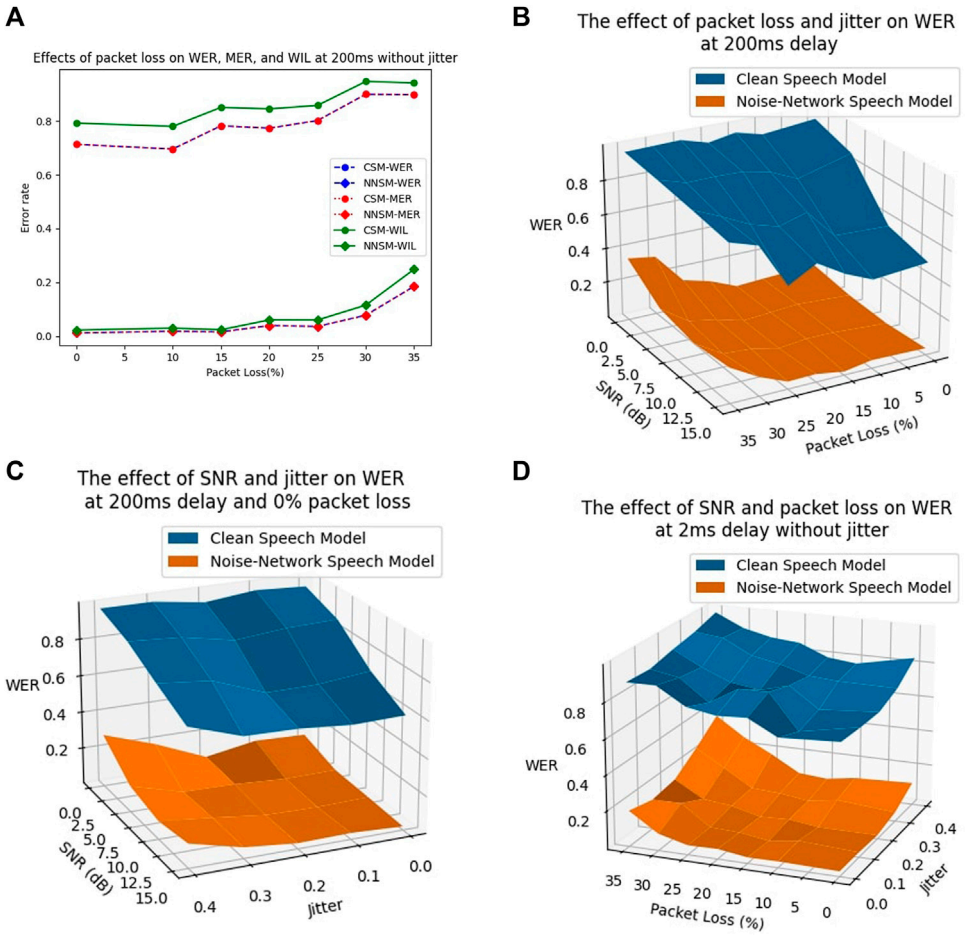

FIGURE 3
The effect of packet loss on WER, MER and WIL, and combination noise-network effects on WER. **(A)** Effects of packet loss on WER, MER, and WIL, for a constant delay of 200 ms without jitter. **(B)** Effects of SNR and packet loss on WER, for a constant delay of 2 ms without jitter. **(C)** Effect of SNR and jitter on WER, for a delay of 200 ms with packet loss of 0%. **(D)** Effects of packet loss and jitter on WER, for a constant delay of 200 ms without jitter.

when the ASR is trained using noise-network speech distorted dataset, the robustness of the ASR increases. The effect of SNR and jitter is observed for the SNR less than 5 *dB* and the jitter greater than 30% of the delay, and the SNR effect does not dominate the jitter effect on the noise-network-trained ASR system.

4.7 Combined effect of jitter and packet loss on WER, MER, and WIL

The combined effect of jitter and packet loss has an increased impact on the performance of ASR for WER, MER, and WIL. Figure 3D shows the effect of jitter and packet loss on WER. Training ASR on noise-network distorted speech minimizes the WER. However, the improvement starts to decrease when the jitter is higher than 0.3 of delay and the packet loss is greater than 15%. The NNSM can learn the patterns for jitter and packet loss better than the CSM models.

5 Conclusion

For VoIP transcription or any other ASR that translates noisy speech transmitted through IP network into text, the ASR model trained on noise-network distorted speech performs better than the clean-speech-trained model. The ASR model trained on noise-network distorted speech can tolerate a jitter of less than 20% and a packet loss of less than 15% without a decrease in the performance. These results are based on

G.722 speech codec without any jitter buffer algorithms and packet loss concealment support. In the next study we will extend this study to include highly versatile codecs like Opus codec Valin et al. (2012) which can scale from low bitrate narrowband speech to fullband speech, support for packet loss concealment and jitter buffer algorithms.

In this study, the dataset includes 30 sentences, which covers all phonemes in the American English language. However, this dataset is small, which results in a degradation on generalization performance. In future studies, a large dataset with a rich set of utterances and speakers can be considered in order to improve the generalization performance of the ASR model trained on noise-network distorted speech dataset. The training method should also be improved in order to maintain generalization performance while learning noise-network distortion features.

It should be noted that the proposed model does not consider the effect of degrading talking or conversation quality. These include response delay, side-tone, talker-echo or any other two-way interaction features.

This study provides an overview of the effect of noise distortions and VoIP-transmission-induced distortions on speech when used as input to ASR. The results of this study can help with network planning for VoIP transcription applications or the deployment of ASR systems, where speech is captured in noisy environments and the transcription is performed remotely.

Data availability statement

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Author contributions

EK, Main researcher, YN PhD supervisor.

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.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Ardila, R., Branson, M., Davis, K., Kohler, M., Meyer, J., Henretty, M., et al. (2020). "Common voice: A massively-multilingual speech corpus," in Proceedings of The 12th Language Resources and Evaluation Conference (Marseille, France: European Language Resources Association), 4218–4222.

Barker, J., Marxer, R., Vincent, E., and Watanabe, S. (2015). "The third 'chime' speech separation and recognition challenge: Dataset, task and baselines," in 2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU), 504–511. doi:10.1109/ASRU.2015.7404837

Barker, J., Vincent, E., Ma, N., Christensen, H., and Green, P. (2013). The pascal chime speech separation and recognition challenge. *Comput. Speech Lang.* 27, 621–633. doi:10.1016/j.csl.2012.10.004

Barker, J., Watanabe, S., Vincent, E., and Trmal, J. (2018). The fifth 'CHiME' speech separation and recognition challenge: Dataset, task and baselines. *Proc. Interspeech* 2018, 1561–1565. doi:10.21437/Interspeech.2018-1768

Brown, K. L., and George, E. B. (1995). Ctimit: A speech corpus for the cellular environment with applications to automatic speech recognition. *ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc.* 1, 105–108. doi:10.1109/icassp.1995.479284

Charniak, E., Blaheta, D., Ge, N., Hall, K., Hale, J., and Johnson, M. (2000). *Bllip 1987-89 wsj corpus release 1*. Philadelphia: Linguistic Data Consortium, 36.

da Silva, A. P. C., Varela, M., de Souza e Silva, E., Leão, R. M., and Rubino, G. (2008). Quality assessment of interactive voice applications. *Comput. Netw.* 52, 1179–1192. doi:10.1016/j.comnet.2008.01.002

Furui, S., Maekawa, K., and Isahara, H. (2000). "A Japanese national project on spontaneous speech corpus and processing technology," in ASR2000-Automatic Speech Recognition: Challenges for the new Millenium ISCA Tutorial and Research Workshop.

Garofolo, J., Lamel, L., Fisher, W., Fiscus, J., Pallett, D., Dahlgren, N., et al. (1993). Timit acoustic-phonetic continuous speech corpus. *Linguist. Data Consort.*

Hannun, A., Case, C., Casper, J., Catanzaro, B., Diamos, G., Elsen, E., et al. (2014). Deep Speech: Scaling up end-to-end speech recognition. *arXiv Prepr. arXiv1412.5567*.

Hu, Y., and Loizou, P. C. (2007). Subjective comparison and evaluation of speech enhancement algorithms. *Speech Commun.* 49, 588–601. doi:10.1016/j.specom.2006.12.006

Hu, Z., Yan, H., Yan, T., Geng, H., and Liu, G. (2020). Evaluating qoe in voip networks with qos mapping and machine learning algorithms. *Neurocomputing* 386, 63–83. doi:10.1016/j.neucom.2019.12.072

Huang, J. T., Li, J., Yu, D., Deng, L., and Gong, Y. (2013). "Cross-language knowledge transfer using multilingual deep neural network with shared hidden layers," in 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, 7304–7308. doi:10.1109/ICASSP.2013.6639081

Hubert, B. (2001). *tc(8) - Linux manual page*.

ITU-T (2005). *G.722.1,"low-complexity coding at 24 and 32 kbit/s for hands-free operation in systems with low frame loss"*. Geneva, Switzerland: International Telecommunication Union.

Kermanshahi, M. A., Akbari, A., and Nasersharif, B. (2021). "Transfer learning for end-to-end asr to deal with low-resource problem in Persian language," in 2021 26th International Computer Conference (Tehran: Computer Society of Iran), 1–5. doi:10.1109/CSICC52343.2021.9420540

Köhn, A., Stegen, F., and Baumann, T. (2016). "Mining the spoken Wikipedia for speech data and beyond," in Proceedings of the Tenth International Conference on Language Resources and Evaluation (Portorož, Slovenia: European Language Resources Association), 4644–4647.

Kornblith, S., Shlens, J., and Le, Q. V. (2019). "Do better imagenet models transfer better?" in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.

Kurematsu, A., Takeda, K., Sagisaka, Y., Katagiri, S., Kuwabara, H., and Shikano, K. (1990). Atr Japanese speech database as a tool of speech recognition and synthesis. *Speech Commun.* 9, 357–363. doi:10.1016/0167-6393(90)90011-W

Li, B., and Sim, K. C. (2014). A spectral masking approach to noise-robust speech recognition using deep neural networks. *IEEE/ACM Trans. Audio Speech Lang. Process.* 22, 1296–1305. doi:10.1109/TASLP.2014.2329237

Linux Foundation (2021). *networking:netem [Wiki]*. San Francisco, California: Linux Foundation.

Malik, M., Malik, M. K., Mehmood, K., and Makhdoom, I. (2021). Automatic speech recognition: A survey. *Multimed. Tools Appl.* 80, 9411–9457. doi:10.1007/s11042-020-10073-7

Morris, A. C., Maier, V., and Green, P. (2004). "From WER and RIL to MER and WIL: Improved evaluation measures for connected speech recognition," in Proc. Interspeech 2004, 2765–2768. doi:10.21437/Interspeech.2004-668

Mozilla (2020). *DeepSpeech 0.9.3*.

Panayotov, V., Chen, G., Povey, D., and Khudanpur, S. (2015). "Librispeech: An asr corpus based on public domain audio books," in 2015 IEEE International Conference on Acoustics, Speech and Signal Processing, 5206–5210. doi:10.1109/ICASSP.2015.7178964

Ravanelli, M., Cristoforetti, L., Gretter, R., Pellin, M., Sosi, A., and Omologo, M. (2015). "The dirha-English corpus and related tasks for distant-speech recognition in domestic environments," in 2015 IEEE Workshop on Automatic Speech Recognition and Understanding, 275–282. doi:10.1109/ASRU.2015.7404805

Rousseau, A., Deléglise, P., and Estève, Y. (2012). "Ted-lium: An automatic speech recognition dedicated corpus," in *Proceedings of the eight international conference on language resources and evaluation*. Editors N. C. C. Chair, K. Choukri, T. Declerck, M. U. Doğan, B. Maegaard, J. Mariani, et al. (Istanbul, Turkey: European Language Resources Association).

Shi, L., Bao, F., Wang, Y., and Gao, G. (2018). "Research on transfer learning for khalkha Mongolian speech recognition based on tdnn," in 2018 International Conference on Asian Language Processing, 85–89. doi:10.1109/IALP.2018.8629237

Sun, L., and Ifeachor, E. (2002). "Perceived speech quality prediction for voice over ip-based networks," in 2002 IEEE International Conference on Communications Conference Proceedings, 2573–2577. vol. 4. doi:10.1109/ICC.2002.997307

Sun, L., and Ifeachor, E. (2006). Voice quality prediction models and their application in voip networks. *IEEE Trans. Multimed.* 8, 809–820. doi:10.1109/TMM.2006.876279

The FFmpeg developers (2020). *FFmpeg documentation*.

Valin, J.-M., Vos, K., and Terriberry, T. (2012). "Definition of the opus audio codec," in *IETF RFC 6716*.

Wang, H. M., Chen, B., Kuo, J. W., and Cheng, S. S. (2005). "Matbn: A Mandarin Chinese broadcast news corpus," in *International journal of computational linguistics & Chinese language processing, volume 10, number 2, june 2005: Special issue on annotated speech corpora*, 219–236.