Improving Low Resource Turkish Speech Recognition with Data Augmentation and TTS

Ramazan Gokay
Speech and Language Technologies Laboratory
TUBITAK BILGEM
Kocaeli, Turkey
ramazan.gokay@tubitak.gov.tr

Hulya Yalcin
Visual Intelligence Laboratory
Istanbul Technical University
Istanbul, Turkey
hulyayalcin@itu.edu.tr

***Abstract*—One of the major problems faced by speech recognition researchers is the lack of data. In this paper, our objective is to compare alternative solutions to lack of data. Some experiments are conducted with very limited training data to see the effects of data augmentation and speech synthesis on speech recognition. Speed and volume perturbations are applied in this study. Besides data augmentation, synthetic speech is generated by using two different speech synthesis methods. In first speech synthesis approach, Google Translate Text to Speech (gTTS) is used as speech synthesizer. In second speech synthesis approach, an end-to-end Turkish TTS system is trained by us. Finally, we examined the effects of all these alternative methods on speech recognition for low resource languages.**

Our results demonstrate that some data augmentation or speech synthesis techniques work well to improve speech recognition for low resource languages. In this study, 14.8% relative Word Error Ratio (WER) improvement is obtained by using combination of augmented and synthetic data.

***Index Terms*—speech recognition, data augmentation, speech synthesis, low resource languages**

I. Introduction

In recent years, the artificial intelligence and machine learning concepts have become quite crucial in human life. Various machine learning tasks are used widely in many areas such as business, finance, social media, security systems.

Automatic speech reconition (ASR) which is the system that converts speech signals into writing whereas text to speech (TTS) which is the opposite of ASR converts text data to speech. Both systems have become important part of human life within developed technology. For example, even smartphones which are carried by many people everyday have ASR and TTS systems. These systems are used for information services, chat-bots, dialog systems or smart assistants sizably.

Deep learning techniques have been applied on ASR and TTS systems within increasing computing power. Some hopeful results were acquired on ASR and TTS system by using deep learning methods. However, these deep learning techniques are pretty data-hungry. They require large amount of data to get better results. In ASR area, significant amount of transcribed audio data is needed to get better results [1] [2]. This situation increases the importance of labeled data. Therefore, significant amount of labor and money are spent to acquire labeled data. In some situations, even the labor and money also may not be enough, hence people try to seek alternative ways. In fact, recently this has been relatively hot topic and it is called ”speech recognition for low resource languages (LRL)”.

NN needs large amount of data

In this study, different data augmentation types and speech synthesiss are examined to improve automatic speech recognition with limited data source. The rest of paper is organised as follows. Section II introduces some literature studies. Section III explains experimental setups. Section IV analyzes the experiment results. Section V concludes the study and discusses ideas for the future work.

II. Related Works

One of the biggest problem faced by ASR researchers mostly is the lack of the properly labeled data. Recently, automatic speech recognition for low resource language (LRL) data has been drawing attention. Indeed, IARPA's BABEL competition is organised to improve speech recognition for limited dataset [3]. There are some approaches to hadle limited data. These approaches rely on data augmentation [1] [4] [5] [6] or speech synthesis [3] [7] techniques mostly.

In [1], data augmentation is applied to speech recognition for limited Assamase and Zulu languages successfully. Vocal tract length perturbation and stochastic feature mapping methods are used for data augmentation. It is experienced that combining of these two methods can improve the ASR result.

Vocal tract length perturbation and stochastic feature mapping methods are used in [4] as well. This data augmentation methods are applied to train acoustic models for Assamese and Haitian Creole languages. The effectiveness of data augmentation is reported in this study.

The approach in [5] demonstrates that augmenting the data at three differing speeds (0.9, 1.0, 1.1 respectively) showed an average of 4.3% WER improvement with the speed perturbation method.

In [3], speech synthesis is used to improve speech recognition and it is emphasized that there is a great potantiel in using synthetic speech to improve ASR results.

978-1-7281-1820-8/19/$31.00 ©2019 IEEE

Speech synthesis is used in [7] as well. According to [7], natural datasets can be augmented within synthetic speech in order to improve accuracy. In the study, 500 hours of synthetic speech was added to 500 hours of natural speech and the WER decreased from 21.31 to 19.54.

III. Experimental Setup

A. Data

In this paper, the experiments are conducted on Turkish speech data. For speech synthesizaton training part, Turkish microphone data read by a professional speaker is used. The recorded microphone data was 16 bit, PCM and 16kHz in the wav file format. The TTS training data is collected from single speaker and the total amount of data is about 18 hours.

For speech recognition part, Turkish news data is used. That ASR data is also subset of the data used in [8]. This data has been collected at Bogazici University within the scope of a TUBITAK research project and it includes broadcast news and TV channel news [8]. In this study, the ASR data is 16 bit, PCM and 16kHz in the wav file format. For baseline, the amount of ASR training data is 10 hours and the amount of ASR test data is 1 hour.

B. Data Augmentation

Data augmentation is a technique which tries to increase the data amount and acoustic variety. The aim of the data augmentation is to incease performance or robustness of the ASR system. There are different types of augmentation techniques such as vocal-tract length perturbation, speed and volume perturbation, noise injection etc. In this study, speed perturbation, volume perturbation and combination of them are used as data augmentation as shown in Fig.1.

Speed Perturbation: The speed of train data is changed according to uniformly random selected coefficient in range 0.85 and 1.15.

Volume Perturbation: The volume of train data is changed according to uniformly random selected coefficient in range -6 and 8.

Combo Perturbation: Speed and volume perturbations are applied to training data at the same time.

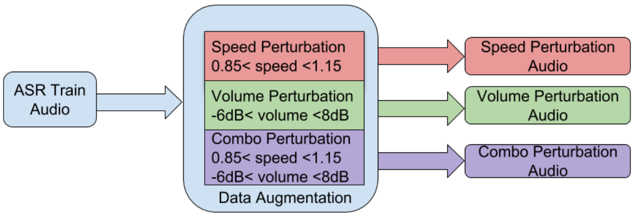
ASR Train Audio
Speed Perturbation 0.85< speed <1.15
Volume Perturbation -6dB< volume <8dB
Combo Perturbation 0.85< speed <1.15 -6dB< volume <8dB
Data Augmentation
Speed Perturbation Audio
Volume Perturbation Audio
Combo Perturbation Audio

Fig. 1. Data Augmentation Approach

C. Speech Synthesis

Speech synthesis is actually a kind of data augmentation technique. In this study, two different approaches are used for speech synthesis as shown in Fig.2. In first approach, Google Translate Text to Speech (gTTS) is used as speech synthesizer. In second approach, an end-to-end Turkish TTS system is trained based on Deep Convolutional TTS (DCTTS) architecture [9] by using approximately 18 hours data. In this part of study, an open source TensorFlow implementation [10] was used. DCTTS system includes two networks as shown in the Fig.3. The first one is called as Text2Mel which synthesizes mel spectogram acording to text input and the second one is called as Spectogram Super Resolution Network (SSRN) which converts mel spectogram to Short Time Fourier Transform (STFT) spectogram [9].

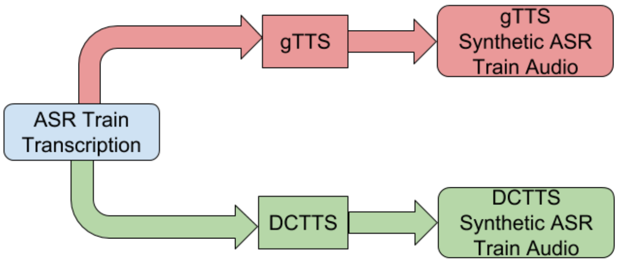
gTTS
gTTS Synthetic ASR Train Audio
ASR Train Transcription
DCTTS
DCTTS Synthetic ASR Train Audio

Fig. 2. Speech Synthesis Approach

In speech synthesis part, ASR training transcriptions were used to get synthetic data. 7000 ASR training utterances which corresponds to approximately 10 hours audio data were synthesized by using each TTS methods.

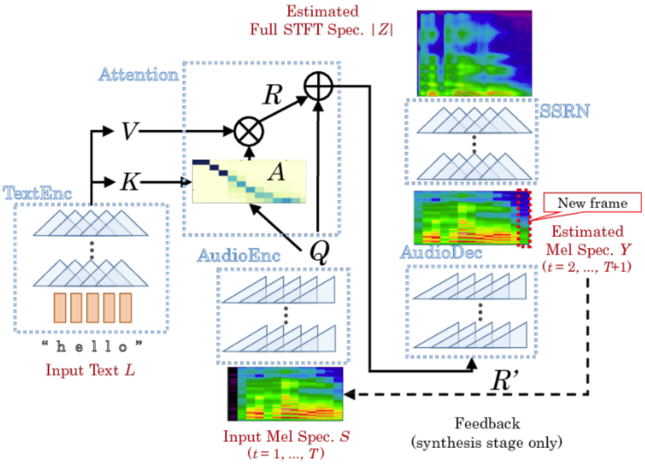
Estimated Full STFT Spec. |Z|
Attention
R
SSRN
V
K
A
TextEnc
New frame
Estimated Mel Spec. Y (t = 2, ..., T+1)
AudioEnc
Q
AudioDec
"h e l l o"
Input Text L
R'
Input Mel Spec. S (t = 1, ..., T)
Feedback (synthesis stage only)

Fig. 3. DCTTS Architecture [9]

D. Speeh Recognition

The experiments were conducted by using end-to-end speech recognition system which was based on Deep

Speech2 architecture [11]. In ASR part of the study, an open source PyTorch implementation [12] of the system was used.

The architecture of used ASR system consists of 2 2D invariant convolutional layers, 7 bidirectional Gated Recurrent Unit (GRU) layers and 1 fully connected layer as shown in Fig. 4 and each GRU layer includes 768 hidden units. The ASR system was trained by using Connectionist Temporal Classification (CTC) loss function [13].

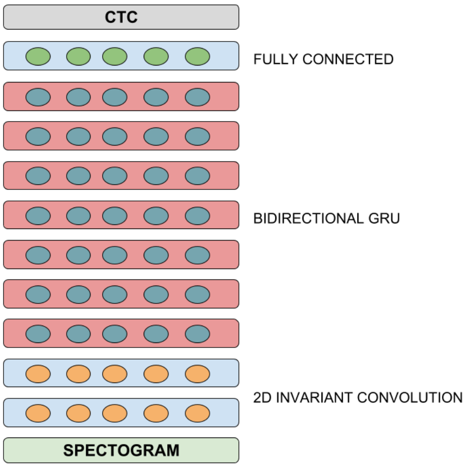

Fig. 4. Deep Speech 2 Architecture used in our approach

In ASR test time, the decoding can be applied by using just the CTC model itself that is called as Greedy Search. Also, CTC model can be coupled with a language model that is called as Beam Search. In this paper, the language model that includes approximately 190 000 words and approximately 3 million 3-grams was generated by using KenLM toolkit [14]. This 3-gram language model was used in beam search decoding with respect to (1).

$$Q(y) = log(p_{ctc}(y|x)) + \alpha log(p_{lm}(y)) + \beta word\_count(y) \quad (1)$$

According to (1), the aim is to find transcription y which maximizes $Q(y)$. The coeffcient α controls the relative contributions of the language model and the CTC network [11]. The coefficient β controls the number of words in the transcription [15]. α and β parameters are fine-tuned to get optimal transcription.

IV. Results and Discussion

In this paper, some experiments are conducted to evaluate the effects of a variety of augmentation types and speech syhthetise methods on automatic speech recognition for low resourced language. For this purpose, 10 different experiments were conducted as shown in Fig.5 and the experiments details are shown in TABLE I.

The first experiment was conducted on baseline condition. In this condition, 10 hours of training data and 1 hour test data were used.

In second experiment, the speed perturbation was applied on the ASR training data to test the speed augmentation effect. By adding only speed augmentated data, the WER decreased from 43.525 to 39.373. Therefore, 9.5% relative WER improvement was obtained through speed perturbation.

In third experiment, the purpose was to see the volume perturbation effect on ASR training. By adding only volume augmentated data, the WER decreased from 43.525 to 39.970. Therefore, 8.2% relative WER improvement was obtained through volume perturbation.

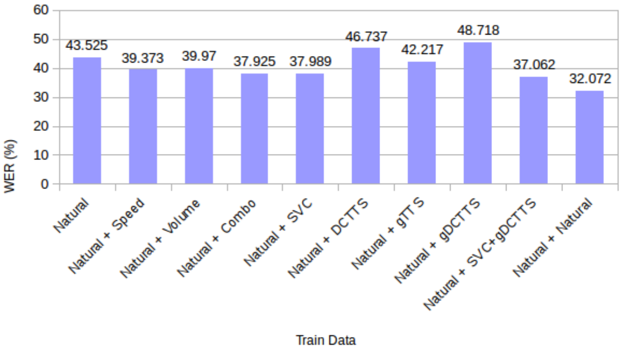

Fig. 5. Experimental Results

In fourth experiment, the both speed and volume perturbation were applied on ASR training data. Combo perturbation was obtained by applying the speed and the volume augmentation at the same time as mentioned earlier. By adding only both augmentated data, the WER decreased from 43.525 to 37.925. Therefore, 12.9% relative WER improvement was obtained through both perturbation.

In fifth experiment, all above augmentated versions were put together to be called as SVC(speed, volume, combo). By adding SVC augmentated data, the WER decreased from 43.525 to 37.989. Therefore, 12.7% relative WER improvement was obtained through SVC augmentation.

TABLE I
AUGMENTATION TYPES COMPARISON

Experiment	Training Data	Duration of Data (hours)	WER(%)
Exp1	Natural (Baseline)	10	43.525
Exp2	Natural + Speed Augment	10+10	39.373
Exp3	Natural + Volume Augment	10+10	39.970
Exp4	Natural + Combo Augment	10+10	37.925
Exp5	Natural + SVC Augment	10+30	37.989
Exp6	Natural + DCTTS	10+10	46.737
Exp7	Natural + gTTS	10+10	42.217
Exp8	Natural + gDCTTS	10+20	48.718
Exp9	Natural + SVC+gDCTTS	10+30+20	37.062
Exp10	Natural + Natural	10+10	32.072

In sixth experiment, the goal was to investigate the speech synthesis effect on ASR training. To do that, a

Turkish TTS model was trained based on DCTTS architecture. In this experiment, single speaker model was generated, hence there was no speaker variation in the synthetic data. By adding single speaker synthesized data, the WER increased from 43.525 to 46.737. Therefore, 7.4% relative WER worsening was monitored through DCTTS synthesized data. This worsening is likely due to the low quality synthetic data. This synthetic data could lead the system to learn the acoustic spesifications incorrectly.

In seventh experiment, gTTS was used instead of our TTS. By adding gTTS synthesized data, the WER decreased from 43.525 to 42.217. Therefore, 3% relative WER improvement was obtained through gTTS synthesized data.

In eighth experiment, all synthesized data was combined to be called gDCTTS (gTTS and DCTTS). By adding all synthesized data, the WER increased from 43.525 to 48.718. Therefore, 11.9% relative WER worsening was monitored within synthesized data which was unpectedly high. Due to the bad synthetic data, the ASR system might have learnt the more frequent letters or charactersof the test data incorrectly and this situation might have led to WER worsening.

In ninth experiment, all above augmented and synthesized versions were combined. By adding augmented and synthetic data, the WER decreased from 43.525 to 37.062. Therefore, 14.8% relative WER improvement was obtained within augmentation and syhthesization combination. This was a somewhat unexpected result. While the combined augmentation improves the WER result, the combined synthesis makes the WER result worser, hence the combination of augmentation and synthesis was expected to give an intermediate result. However, the combination of the augmentation and the synthesis provided the best WER result in artificially generated data category. This situation may be explained with large amount of training data and the increment of the acoustic variation.

In the final experiment, the natural ASR traing data was doubled. By adding natural data, the WER decreased from 43.525 to 32.072. Therefore, 26.3% relative WER improvement that was the best WER result was obtained through adding natural data.

According to above experiment results, the best improvement was obtained by adding natural data as expected. However, if there is no avaliable natural data, some augmentation or synthesis techniques can be applied to get better results.

V. Conclusion and Future Work

In this study, alternative solutions to the lack of data, one of the biggest problems faced by speech recognition researchers were examined. Some experiments were conducted with very limited training data to see the effects of data augmentation and speech synthesis on speech recognition.

Our results showed that some augmentation or synthesis techniques work well to improve speech recognition for low resource language. In our case, 14.8% relative WER improvement was obtained by using combination of augmented and synthetic data. However, it is a known fact that adding artificial data is not as effective as adding natural data.

In our work, we covered only Turkish television data. As a future work, it can be extended to other languages or telephone domain. Also, another TTS can be applied for speech synthesis.

References

[1] A. Ragni, K. M. Knill, S. P. Rath, and M. J. Gales, "Data augmentation for low resource languages," in *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

[2] G. Evermann, H. Y. Chan, M. J. Gales, T. Hain, X. Liu, D. Mrva, L. Wang, and P. C. Woodland, "Development of the 2003 cu-htk conversational telephone speech transcription system," in *Acoustics, Speech, and Signal Processing, 2004. Proceedings.(ICASSP'04). IEEE International Conference on*, vol. 1. IEEE, 2004, pp. I–249.

[3] L. V. Rygaard, "Using synthesized speech to improve speech recognition for low–resource languages."

[4] X. Cui, V. Goel, and B. Kingsbury, "Data augmentation for deep neural network acoustic modeling," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 23, no. 9, pp. 1469–1477, 2015.

[5] T. Ko, V. Peddinti, D. Povey, and S. Khudanpur, "Audio augmentation for speech recognition," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[6] I. Rebai, Y. BenAyed, W. Mahdi, and J.-P. Lorré, "Improving speech recognition using data augmentation and acoustic model fusion," *Procedia Computer Science*, vol. 112, pp. 316–322, 2017.

[7] F. F. A. Bonab and S. Ginn, "Learning to recognize speech from chaotically synthesized data."

[8] E. Arısoy, "Statistical and discriminative language modeling for turkish large vocabulary continuous speech recognition," *Phd Thesis, Bogazici University*, 2009.

[9] H. Tachibana, K. Uenoyama, and S. Aihara, "Efficiently trainable text-to-speech system based on deep convolutional networks with guided attention," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4784–4788.

[10] Kyubyong, "A tensorflow implementation of dc-tts," (accessed October 30, 2018). [Online]. Available: https://github.com/Kyubyong/dc_tts

[11] D. Amodei, S. Ananthanarayanan, R. Anubhai, J. Bai, E. Battenberg, C. Case, J. Casper, B. Catanzaro, Q. Cheng, G. Chen *et al.*, "Deep speech 2: End-to-end speech recognition in english and mandarin," in *International Conference on Machine Learning*, 2016, pp. 173–182.

[12] SeanNaren, "Speech recognition using deepspeech2," (accessed October 30, 2018). [Online]. Available: https://github.com/SeanNaren/deepspeech.pytorch

[13] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*. ACM, 2006, pp. 369–376.

[14] K. Heafield, I. Pouzyrevsky, J. H. Clark, and P. Koehn, "Scalable modified kneser-ney language model estimation," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, vol. 2, 2013, pp. 690–696.

[15] N. Tomashenko and Y. Esteve, "Ted-lium 3: Twice as much data and corpus repartition for experiments on speaker adaptation," in *Speech and Computer: 20th International Conference, SPECOM 2018, Leipzig, Germany, September 18–22, 2018, Proceedings*. Springer, p. 198.

Summary:

Synthesized data can be appended into training data to Improve the lack of data. but the quality is not as good as natural language signal.