Data Selection in the Framework of Automatic Speech Recognition

Ismael Bada, Juan Karsten, Dominique Fohr, Irina Illina

Data Selection in the Framework of Automatic Speech Recognition

Ismael Bada, Juan Karsten, Dominique Fohr, Irina Illina

MultiSpeech team
Université de Lorraine, LORIA, UMR 7503, Vandoeuvre-lès-Nancy, F-54506, France
Inria, Villers-lès-Nancy, F-54600, France
CNRS, LORIA, UMR 7503, Vandoeuvre-lès-Nancy, F-54506, France

Abstract

Training a speech recognition system needs audio data and their corresponding exact transcriptions. However, manual transcribing is expensive, labor intensive and error-prone. Some sources, such as TV broadcast, have subtitles. Subtitles are closed to the exact transcription, but not exactly the same. Some sentences might be paraphrased, deleted, changed in word order, etc. Building automatic speech recognition from inexact subtitles may result in a poor models and low performance system. Therefore, selecting data is crucial to obtain a highly performance models. In this work, we explore the lightly supervised approach, which is a process to select a good acoustic data to train Deep Neural Network acoustic models. We study data selection methods based on phone matched error rate and average word duration. Furthermore, we propose a new data selection method combining three recognizers. Recognizing the development set produces word error rate that is the metric to measure how good the model is. Data selection methods are evaluated on the real TV broadcast dataset.

Index terms: speech recognition, neural networks, acoustic model, data selection

1. Introduction

Automatic speech recognition (ASR) is one of the sub field of natural language processing with many practical applications: automatic closed captioning for hearing-disabled persons, taking notes of conversations between doctors and patients, voice control and many more. Despite of the rapid development of speech recognition, there are still many challenges in the field. One of the challenges is the training of a speech recognizer, which requires a huge amount of transcribed training data. The transcribed training data consists of audio data and the corresponding text transcriptions. However, transcribing audio manually is labor intensive and also time consuming. There exist many unlimited supply of audio data from internet, TV broadcasts, radio, as well as video streaming websites, but there is no available exact transcription. However, some TV broadcasts, such as *CNN headline news, ABC world news tonight, BBC*, have subtitles that can be used for training a speech recognition system.

To train a speech recognition system, one possibility is to use TV broadcasts data that have subtitles. These subtitles are close, but not exactly the same as what people uttered. Some sentences might be paraphrased, deleted, changed in word order, etc. There are some examples of approximate subtitles:

- Real transcription:
  Russia started badly with the dropping at the hands of Spain. But, they got better and better. Spain looked unstoppable to start with but since then they have looked a little.
- Corresponding subtitle:
  Russia started badly with at beating at the hands of Spain. Spain looked then they have looked a little.

Furthermore, subtitles are often badly aligned with the audio. Some segments in training audio can contain unconstrained conversational speech, use of foreign words, high out-of-vocabulary rates, channel noise and simultaneous speech from more than one speaker. Even, thus audio data is sometimes difficult to be recognized by humans. These facts make hard to use subtitles for ASR.

The idea of using untranscribed audio data (or unsupervised training, no subtitles) has been proposed firstly in [15] and [5]. Authors of [15] proposed an iterative training procedure: decode untranscribed data and keep only the segments with high confidence score for the next training iteration. Even an 80% error rate system can improve itself automatically, but the system performance is limited. [6] were the first to propose *lightly supervised training* with a large amount of training data. Instead of using untranscribed training data, they trained speech recognition system using audio data with subtitles. Lightly supervised approach allows selecting "good" training data. First, an acoustic model from another task (or another corpus) is used to recognize audio data. The decoding results are compared with the subtitles and removed if they disagree. These selected data are used to train a new acoustic model. [3] proposed the confidence measure metric to remove the bad audio segments. When decoding acoustic inputs, an ASR produces word hypothesis and their corresponding confidence measure. The confidence measure value is used to remove potentially bad segments where the confident value is lower than a threshold. [10] applied lightly supervised approach on medical conversation data.

Very recently, a new point of view on the data selection has been proposed. [8] suggest an original two-stage crowdsourcing alternative. First, iteratively collects transcription hypotheses from the web and, then, asks different crowds to pick the best of them. [9] proposed an approach to domain adaptation that does not require transcriptions but instead uses a corpus of unlabeled parallel data, consisting of pairs of samples from the source domain of the well-trained model and the desired target domain.

In the present paper, the same problem of data selection for acoustic modeling training using a huge data corpus is considered. We want to select a good acoustic data to train Deep Neural Network acoustic models. The scientific contributions of this paper are:

- We study the impact of data selection on the word error rate.
- We explore different variations of slightly supervised training of acoustic models.

- We present a comparison of different data selection approaches in the context of TV broadcast news speech transcription.

2. Methodology

2.1. Lightly supervised data selection

To generate an accurate speech recognition, a very large training audio corpus with its exact corresponding transcription is required. This is particularly true for Deep Neural Network (DNN) based systems, having millions of parameters to train. However, transcribing audio is labor intensive and time consuming. There are unlimited supply of audio data in the internet, television, radio and other sources. But very few have available transcription. However, some TV broadcasts have subtitles. By utilizing these audio data with the corresponding subtitles, we hope to produce a high performance speech recognizer with less supervision. Nevertheless, some problems exist when using the data with subtitles as training dataset. Training using the subtitles faces several disadvantages compared to the manual transcriptions: indication of non-speech events (coughing, speaker turn) and acoustic conditions (background noise, music, etc.) are missing.

The main idea of lightly supervised approach is to use the automatic speech recognizer to transcribe training audio data. After this, only well transcribed segments (segments where automatic transcription corresponds to subtitles) will be used as training data [6].

We assume that we have a massive amount of training audio data and corresponding subtitles. In general, the lightly supervised approach operates as follow:

1. Randomly select a subset of the training set.
2. Train an acoustic model on a small amount of manually annotated data or use an acoustic model from another task.
3. Using ASR, recognize all training audio data.
4. Align the automatic transcriptions with the subtitles of the training data. Some transcriptions and subtitles might disagree. We can remove or correct these segments.
5. Retrain a new acoustic model using the data we selected in the previous step.
6. Optionally reiterate from step 3.

These steps can be iterated several times as long as the error rate is decreasing. This method uses the idea of training acoustic models in less supervised manner because the training dataset (subtitles) is not the actual transcription. Using subtitles as training data greatly reduces the manual transcription effort (20-40 time less).

2.2. Revisited lightly supervised data selection

In the lightly supervised approach presented previously, a very important step is the step 3. In the case of a disagreement between automatic transcriptions and subtitles, which part of subtitles to keep and which part to remove or correct? Can we use additional criteria to better choose the training data? How many training data to keep? In this section, we propose to study some of these questions.

2.2.1. Using AWD and PMER

According to [7], using of *Average Word Duration* (AWD) and *Phone Matched Error Rate* (PMER) during the data selection step (step 3) allows increasing greatly the quality of the selected training data. AWD is used as metric to detect if errors occur in aligning the start and end time of a segment or if something went wrong in the recognition process.

$$AWD = \frac{utterance\ duration\ in\ second}{number\ of\ words\ in\ the\ recognized\ utterance}$$

Usually, duration of a word cannot exceed an upper limit threshold and the duration cannot be lower than a bottom limit threshold. If it is the case, this means that the corresponding transcription or subtitle is wrong.

Phone Error Rate (PER) and *Word Error Rate* (WER) are usually used to measure the performance of a speech recognition system:

$$WER = 100 * \frac{substitutions + deletions + insertions}{number\_of\_words}$$

Error rate is obtained by comparing exact transcriptions and decoding transcriptions produced by the speech recognition system. Word error rate is obtained by the comparison at the word level, phone error rate at the phone level. Our training set has only subtitles. So we can only compare subtitles and recognized transcriptions. To avoid the confusion, we will use Phone Matched Error Rate (PMER). High PMER shows that at phone level the corresponding subtitle is very different compared to recognized transcription. This means possible problems in audio signal (noise, music) or in subtitle. In this case, it is better to discard this segment from the training set.

We chose to use PMER and not WMER because we use phone acoustic models. During acoustic training we interested by the phone sequence and not by the word sequence. For example, the words "too" and "two" have the same sequence of phones: */ t uw /*. If we misrecognized "two" instead of "too", it will be sad to reject corresponding subtitle since these two words have a same phone sequence.

In our work, we propose to use these measures to increase the quality of the data selection. The proposed iterative methodology is as follow:

1. Randomly select a subset of the training set. This set is used to train an initial acoustic model.
2. Train an acoustic model using the audio and the subtitles of this training set.
3. Decode the full training set with the obtained acoustic model. This will produce new decoding results and new values of PMER and AWD. The new values of PMER are obtained from comparing the subtitles and the decoding results.
4. Select the subtitles from the training set based on AWD as follow: $threshold_1 < AWD < threshold_2$
5. Sort the obtained segments according to PMER. Choose *N* hours of the top PMER segments to make a new training set to train the next acoustic model.
6. Continue the step 2-5 until the data selection does not improve anymore.

At each iteration, the number *N* of selected hours can be augmented. To measure the improvement of the approach at each iteration, a development set recognition could be performed.

2.2.2. System combination

Usually, different ASR systems (with different acoustic models and/or language models) will make different errors. Thus, if several systems provide the same transcription as the original subtitle for one segment, it is very likely that the subtitle corresponds exactly to what has been uttered. We can use it reliably for training acoustic models.

The general idea of system combination approach is to combine different ASR systems by varying the language models or acoustic models or both. We have chosen to vary the language model because an acoustic model variation is a very time consuming task when we use a huge training set. The language models can be built with different constraints. A constrained

language model is trained only with the sentences of the training corpus used for selection. A less constrained LM is trained also on data from different sources. The idea is as follows: if the recognition result of one ASR and the recognition result of another ASR are the same, we can trust this recognition result. The proposed combination approach works almost in the same way as the method of section 2.3: training acoustic model with the subset of training data (audio data and their subtitles), recognizing and selecting from the full training data and repeating these steps to do better data selection. However, the difference lies in the recognition and data selection steps.

We built three speech recognition systems and each recognition system is used to perform recognition of the same training set. Consequently, we had three transcriptions which have the same amount of segments. We average the value of PMER and AWD from three corresponding decoding transcriptions. After this, we select the training data (step 4 of the approach presented in section 2.2.1) with the proposed combination algorithm:

Select the subtitles from the training set based on AWD as follow: $threshold_1$ < AWD < $threshold_2$

If a segment has zero PMER with one ASR, select the segment and corresponding subtitles

Else

If a segment have the same phone sequence using two ASRs and PMER< $threshold_{PMER}$, select the segment and the corresponding subtitle.

Else sort by average PMER. Choose top N hours segments with the lowest PMER. These subtitles will be chosen to train the next acoustic model.

We hope that using the recognition results when two ASRs agree will help. If a development corpus is available, the thresholds can be chosen to minimize word error rate.

3. Experiments

3.1. Audio corpus

We used the data from the *Multi Genre Broadcast* (MGB) challenge [1], [16]. MGB is a challenge to automatically transcribe TV broadcasts. TV broadcast data are recorded in highly diverse environments, speech with background music, non-speech events and sounds, etc. The challenge organizers only provided TV broadcast audio data and their corresponding subtitles. As presented previously, subtitles may be different compared to the actual transcription due to deletion, insertion, substitution and paraphrasing. Thus, MGB data recognition is a very difficult task.

MGB challenge data consists of:

1. A training set contains audio data with their corresponding subtitles. This training set is used for training speech recognition systems.
2. A development set contains around 8 hours of audio data and their corresponding *manual* transcriptions (exact transcriptions). This dataset can be used to evaluate studied approaches.
3. A text corpus: 640 million words of TV subtitles are provided. These data can be used to train ASR language model.

Datasets	**# of shows**	**# hours of shows**	**# hours of speech**
Training	751	470	349
Development	16	8.8	6.8

Table 1: *MGB challenge datasets.*

Table 1 shows the statistics of the training and the development sets. We can see that, in average, each show contains about 2/3 of speech and 1/3 of non-speech events. These non-speech events are difficult to recognize.

3.2. Transcription system

KATS (*Kaldi-based Automatic Transcription System*) speech recognition system is based on Context Dependent HMM-TDNN phone models [11] [2] [4]. We used Kaldi toolkit for training and for recognition [13]. The TDNN architecture has 6 hidden layers, each hidden unit utilizes Rectified Linear Unit (RELU) activation function. The TDNN has around 9100 output nodes (senones) with *softmax* activation function. The feature vectors are MFCC with 40 feature values. The baseline phonetic lexicon contains 118k pronunciations for 112k words. Using the *pocolm* [12], 3-gram language model is estimated on text corpora of about 640 million words.

4. Experimental results

4.1. Study of MGB training data

Number of segments (subtitles)	253 K
Average segment duration	4.96 sec
Average number of words per segment	14.4
Vocabulary size	52 K
Total number of words	3 650 K

Table 2: *MGB challenge train set statistics.*

Table 2 and figure 1 present some statistics of the training set of MGB. From the figure we observe that the training set has a high number of segments (subtitles) of average duration of 4.96 seconds and about 14.4 words per segment. Figure 1 shows that there is a large number of subtitles with only few words, so they correspond to a very short speech duration. These short segments can be not easy to recognize by an ASR.

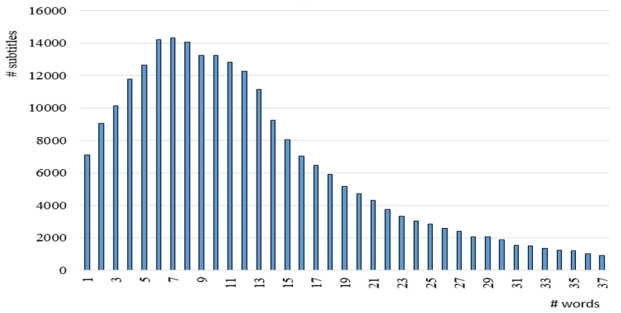

Figure 1: *Histogram of number of words in function of number of subtitles. MGB training set.*

Figure 2 displays the histogram of number of hours of speech in function of AWD for the training set. AWD values were given by the MGB organizers and obtained after ASR recognition. We can see that the majority of speech segments have an AWD between 0.25 seconds and 0.6 seconds. If one segment has a very small AWD or a very high one, it means that something went wrong and this segment corresponds rather to non-speech events. For safety reasons, for data selection we have extended this interval and we chose AWD between 0.16 and 0.6 for the following experiments.

Table 3 presents the number of speech hours in function of PMER on the training set. PMER values for each segment were given by the MGB organizers. We can observe that one third of the training data have PMER greater than 30%. This means that if we want to keep only a very good training data with very low PMER (so, with a very good quality subtitles), we will have a small training set. In contrast, if we keep all data, a large number

of subtitles do not correspond to what was uttered and a negative impact on training is observed.

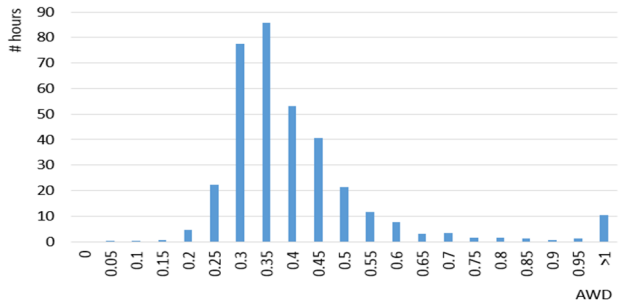

Figure 2: *Distribution of number of hours of speech in function of AWD (in seconds) for the training set*

PMER	<3	<15	<30	<50	<80	All
# hours	112	210	260	304	311	349
Duration %	32	60	74	87	89	100

Table 3: *Number of hours of training speech according to PMER. Duration (%) as percentage of the total train set.*

4.2. Impact of the data selection

In order to assess the influence of the data selection, we trained different ASR systems with different amount of training data. The amount of training hours is selected according to the PMER provided by the MGB organizers. For these experiments, we kept only training data with the PMER below some threshold.

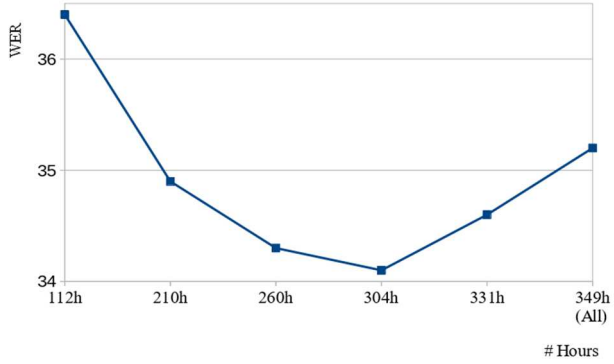

Figure 3: *WER on the development set according to the number of hours selected for training the ASR system.*

Figure 3 presents the WER on the development set according to the number of hours selected for training the ASR system (see Table 3). For example, for PMER below 50 we kept only 305 hours of corresponding training data. The trained system is used to recognize the development data and the obtained WER is presented in Figure 3. From this figure we can observe that, at first, WER decreases when the amount of training data increases. But selecting too much data (i.e. using subtitles with high PMER) the WER begins to increase. Therefore, it is important to find a compromise between the quantity of training data and the quality of training data. In conclusion, data selection is important to train an efficient ASR.

4.3. Results of data selection methods

We studied and evaluated the presented data selection approaches on the development corpus.

The execution time of one iteration of data selection takes about 43 hours. This is very time consuming. To speed up the experiments, at each iteration of data selection, we decided to select a different number of hours of data (parameter *N* in the selection algorithm). We hope that a strong selection at the first iteration and less constrained selection at the next iterations will improve the recognition results.

To build an initial acoustic model (called *ASR-AM0)*, we selected randomly 100 hours because we do not have any information about the quality of available subtitles (in real life, only subtitles are available with no addition information, neither PMER nor AWD).

According to the algorithm of section 2.2.1, during the first iteration of data selection, we decode the whole training corpus with *ASR-AM0* with KATS system. We kept only segments whose AWD is inside [0.16, 0.6]. We excluded all other segments. We sorted the remaining segments according to PMER and we select *N=150* hours. With these 150 hours, we trained *ASR-AM1*. For second iteration *N=200h* (*ASR-AM2*) and for the last iteration *N=300h* (*ASR-AM3*).

Table 4 presents the recognition results on the development set for each iteration of the data selection algorithm. Results are presented in terms of percent of correct recognition and in term of WER. The best results are highlighted in bold. Table shows that each data selection iteration improves the ASR system. The best result of 35% WER is obtained at the last iteration. Performing one more iteration gives the same result and is not presented in the table. Results presented in table 4 are not comparable with those in figure 3 because in figure 3, we used PMER given by the organizers.

ASR	#hours selected	PMER	Corr (%)	WER (%)
ASR-AM0	100	10	65.2	40.2
ASR-AM1	150	15	69.0	36.1
ASR-AM2	200	21	69.3	35.7
ASR-AM3	300	49	69.7	**35.0**

Table 4: *WER on development set for different data selection iterations. Language model is estimated on text corpora of about 640 million words.*

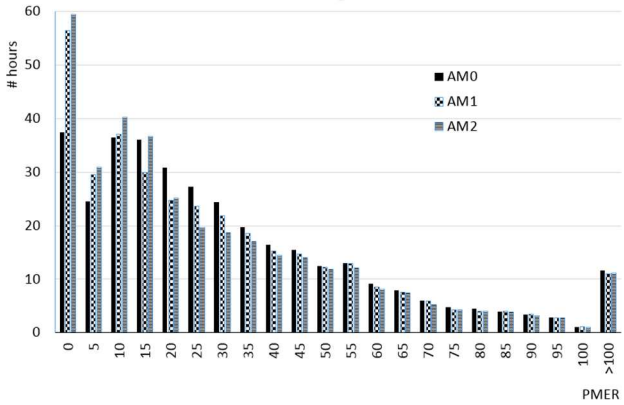

Figure 4: *Number of hours of speech in function of PMER for the train set. For example, to PMER between 5 and 10 corresponds about 36 hours of speech.*

Figure 4 gives more details about the training data distribution in function of PMER values and for different acoustic models. Firstly, this figure shows that a large amount of training data have a PMER below 15 and, so, have good quality subtitles. Secondly, PMER of 0 corresponds about 38 hours of speech at initial iteration, 57 hours for ASR-AM1 and about 60 hours for ASR-AM2. This shows that from one iteration to another the acoustic model performance increases and the acoustic model choose better training data.

System combination

For system combination, we designed three different recognition systems. They share the same acoustic models (*ASR-AM3*), but the language models are different. For the first one only the subtitles of the training corpus are used to train the LM. This model is the most constrained. For the second one, the LM is trained using 640 million words of TV subtitles provided by the

organizers of the MGB challenge (least constrained model). The last one is a combination of the two previous language models. The $threshold_{PMER}$ was chosen experimentally and its value is 30.

Using these three ASRs for system combination, a relative improvement of 2% on WER was observed compared to *ASR-AM3* (cf. table 5). This improvement is significant. It could be interesting to combine systems using different acoustic models, for instance different acoustic features or different neural networks architecture (Long Short Term Memory, Highway networks).

ASR	#hours selected	Corr (%)	(Sub, Del, Ins) (%)	WER (%)
ASR-AM3	300	69.7	(14.3, 16.0, 4.7)	35.0
System combination	300	**70.2**	**(13.8, 16.0, 4.5)**	**34.3**

Table 5: *WER on development set.*

5. Conclusion

In this article, we explored different methods of data selection for building an automatic speech recognition system. The methods are inspired by lightly supervised technique. We studied data selection methods based on phone matched error rate and average word duration. Furthermore, we proposed a new data selection method combining three recognizers. The experiments are conducted on a TV broadcast corpus with subtitles. We have shown that selecting data is crucial for obtaining accurate acoustic models. We have studied the influence of PMER on data selection. The proposed system combination is beneficial to select better data and to obtain an efficient acoustic model.

6. Acknowledgements

This work is funded by the CPER LCNH project supported by the Great East region and by AMIS (Access Multilingual Information opinionS) Chist-Era project. Experiments presented in this paper were carried out using the Grid'5000 testbed, supported by a scientific interest group hosted by Inria, CNRS, RENATER and other Universities and organizations (https://www.grid5000.fr).

7. References

[1] Bell, P., Gales, MJF, Hain, T., Kilgour, J, Lanchantin, P., Liu, X., McParland, A., Renals, S., Saz, O., Wester, M., Woodland, PC. (2015). The MGB Challenge: Evaluating Multi-Genre Broadcast Media Recognition. In Proceedings of IEEE ASRU.

[2] Bengio, Y., Goodfellow, I., Courville, A. (2015). Deep Learning. Book in preparation for MIT Press.

[3] Chan, H. and Woodland, P. (2004). Improving broadcast news transcription by lightly supervised discriminative training. In 2004 IEEE International Conference on Acoustics, Speech, and Signal Processing. ICASSP.

[4] Deng, L., Li, J., Huang, J.-T., Yao, K., Yu, D., Seide, F., Seltzer, M., Zweig, G., He, X., Williams, J., Gong, Y. and Acero A. (2013). Recent Advances in Deep Learning for Speech Research at Microsoft. Proceedings of ICASSP.

[5] Kemp, T. and Waibel, A. (1999). Unsupervised Training of a Speech Recognizer: Recent Experiments. In Proceedings of Eurospeech.

[6] Lamel, L., Gauvain, J.-L., and Adda, G. (2002). Lightly Super-vised and Unsupervised Acoustic Model Training. Journal Computer Speech and Language. 16:115-129.

[7] Lanchantin, P., Gales, M. J., Karanasou, P., Liu, X., Qian, Y., Wang, L., Woodland, P., and Zhang, C. (2016). Selection of multi-genre broadcast data for the training of automatic speech recognition systems. In Proceedings of Interspeech.

[8] Levit, M., Huang, Y., Chang, S. and Gong Y. (2017). Don't Count on ASR to Transcribe for You: Breaking Bias with Two Crowds. In Proceedings of Interspeech.

[9] Li, J., Seltzer, M., Wang, X., Zhao, R., Gong Y. (2017). Large-Scale Domain Adaptation via Teacher-Student Learning. . In Proceedings of Interspeech.

[10] Mathias, L., Yegnanarayanan, G., and Fritsch, J. (2005). Discriminative training of acoustic models applied to domains with unreliable transcripts. In Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP.

[11] Peddinti, V., Povey, D., Khudanpur, S. (2015). A time delay neural network architecture for efficient modeling of long temporal contexts Proceedings of Interspeech.

[12] Povey, Pocolm https://github.com/danpovey/pocolm

[13] Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlicek, P., Qian, Y., Schwarz, P., Silovsky, J.,Stemmer, G., and Vesely, K. (2011). The kaldi speech recognition toolkit. In Proceedings of IEEE 2011 Workshop on Automatic Speech Recognition and Understanding (ASRU).

[14] Stolcke, A. (2002). SRILM - An Extensible Language Modeling Toolkit. Proceedings of ICSLP.

[15] Zavaliagkos, G. and Colthurst, T. (1998). Utilizing Untranscribed Training Data to Improve Performance. DARPA Broadcast News Transcription and Understanding workshop.

[16] Woodland, P. C., Liu, X., Qian, Y., Zhang, C., Gales, M. J. F., Karanasou, P., Lanchantin, P., and Wang, L. (2015). Cambridge University Transcription Systems for the Multi-genre Broadcast Challenge. In IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU).