INTERSPEECH
2009 BRIGHTON

PodCastle: Collaborative Training of Acoustic Models on the Basis of Wisdom of Crowds for Podcast Transcription

Jun Ogata and Masataka Goto

National Institute of Advanced Industrial Science and Technology (AIST)
1-1-1 Umezono, Tsukuba, Ibaraki 305-8568, JAPAN

Abstract

This paper presents acoustic-model-training techniques for improving automatic transcription of podcasts. A typical approach for acoustic modeling is to create a task-specific corpus including hundreds (or even thousands) of hours of speech data and their accurate transcriptions. This approach, however, is impractical in podcast-transcription task because manual generation of the transcriptions of the large amounts of speech covering all the various types of podcast contents will be too costly and time consuming. To solve this problem, we introduce collaborative training of acoustic models on the basis of wisdom of crowds, i.e., the transcriptions of podcast-speech data are generated by anonymous users on our web service PodCastle. We then describe a podcast-dependent acoustic modeling system by using RSS metadata to deal with the differences of acoustic conditions in podcast speech data. From our experimental results on actual podcast speech data, the effectiveness of the proposed acoustic model training was confirmed.

Index Terms: podcast, LVCSR, acoustic model training, wisdom of crowds, error correction

1. Introduction

One of the main applications of the automatic speech recognition of multimedia data is to serve as the basis for automatic indexation for an information retrieval system. From this viewpoint, spoken document retrieval (SDR) systems have been developed by many research groups across the world [1][2]. However, SDR systems or technologies have not been in practical use such as text retrieval engines (e.g., Google and Yahoo!). because the automatic transcription and indexation of multimedia data in the real world is very difficult. The amount of speech data is increasing rapidly on the web because *podcasts* that are often referred to as audio blogs have recently become popular and widespread. We therefore think SDR technologies and systems are in great demand especially for web applications or services. As an application of SDR technologies based on speech recognition, we have developed a web application, called *"PodCastle"*[3][4][5] that allows a number of anonymous users to search and read podcasts, and to share the full text of speech recognition results for podcasts. In order to improve the usefulness of the web service, we have studied and developed several speech recognition methods for accurate podcast transcription [4].

Recent advances in the state-of-the-art large vocabulary continuous speech recognition systems [6]-[8] can be attributed to the availability of large amounts of training data. In general, these training data are collected for each specific task like broadcasting news, lectures, meetings, etc., and their accurate transcriptions are manually generated. On the other hand, the acoustic conditions in podcasts, such as speaker characteristics, channel type, and background noise, are diverse. Therefore, adopting a common approach based on a task-specific corpus is impractical because in podcast transcription, it will be too costly and time consuming to prepare a corpus that covers all the content and the acoustic conditions in podcast transcription tasks.

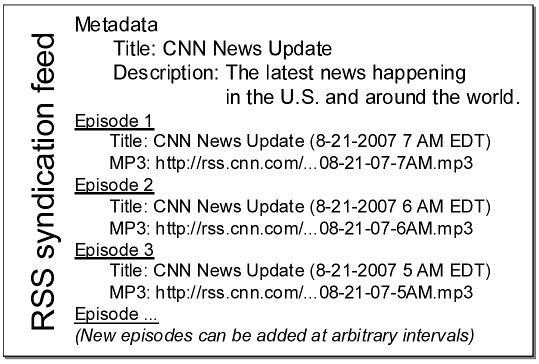
In this paper, we focus on acoustic-model-training techniques to improve the recognition performances in podcast transcription. In order to prepare the training data in podcast transcription task, we study the use of the benefits of our web service PodCastle. PodCastle encourages users to cooperate by correcting speech-recognition errors so that podcasts can be searched more reliably. If a lot of users contribute to the error correction, PodCastle can provide relatively accurate transcriptions of podcasts. We therefore incorporate such *wisdom of crowds* from the users' contributions into acoustic model training, i.e., the transcriptions generated by the anonymous users are used for acoustic model training. Furthermore, we examine podcast-dependent acoustic models making use of podcasting mechanisms.

The remainder of the paper is organized as follows. In the next section, we present a brief explanation of podcasts and several problems in podcast transcription. We then present the acoustic model training method based on the wisdom of crowds, and Section 4 gives some experimental results. In the final section, we state our conclusions and future plans.

2. Automatic Transcription of Podcasts

Podcasting is a mechanism in which multimedia files distributed over the web can be downloaded automatically for playback on portable media players and personal computers, and it has become very popular recently. A podcast consists of several audio data (MP3 files) called *episodes* and a syndication feed (RSS) that includes metadata information about episodes as shown in Figure 1. The feed provides not only the list of URLs of audio files by which episodes can be accessed but also other information such as the published date, titles, and summaries. With regard to podcasting, the amount of audio data on the web has been steadily increasing.

Even state-of-the-art speech recognition systems have difficulties in transcribing podcasts because their contents and recording environments vary very widely. As a problem related to language modeling, podcasts tend to include words and phrases related to recent topics, which are usually not registered in the system vocabulary. We have tackled this problem by using a method to keep a language model up-to-date by using on-line news texts [4]. On the other hand, as a problem related to acoustic modeling, which is the focus of this work, podcasts include various types of speech data, for example, pure



Metadata
Title: CNN News Update
Description: The latest news happening in the U.S. and around the world.
Episode 1
Title: CNN News Update (8-21-2007 7 AM EDT)
MP3: http://rss.cnn.com/...08-21-07-7AM.mp3
Episode 2
Title: CNN News Update (8-21-2007 6 AM EDT)
MP3: http://rss.cnn.com/...08-21-07-6AM.mp3
Episode 3
Title: CNN News Update (8-21-2007 5 AM EDT)
MP3: http://rss.cnn.com/...08-21-07-5AM.mp3
Episode ...
(New episodes can be added at arbitrary intervals)

Figure 1: Example of a syndication feed (RSS) for a podcast.

Table 1: The number of podcasts, episodes, and episodes corrected by anonymous users as of January 26th, 2009.

# registered podcasts	482
# registered episodes (mp3 files)	37825
# corrected episodes	1489

speech, noisy speech, narrow-band speech, and speech with music. Moreover, speaking styles also vary depending on the subdomains like news, lectures, chitchat shows, etc. Even though we have dealt with this issue by applying several improvement methods such as noise suppression at the front-end [9] and iterative unsupervised MLLR adaptation [15], the recognition performance in the podcast transcription task was not very good [4].

3. Acoustic Model Training Based on Wisdom of Crowds

In this section, we present an acoustic-model-training system based on recognition-error corrections that are made by anonymous users of PodCastle.

3.1. Recognition-error Correction by Anonymous Users

PodCastle provides an error correction interface in which recognition errors can be quickly and easily corrected by users [10]. For each audio data of a podcast episode, this function displays not only a 1-best word sequence but also a numbered list of N-best competitive candidates for each word (not for a sentence) as shown in Figure 2. A user who finds a recognition error can simply select the correct word from the candidates. When the correct word is not shown in the candidates, the user can also correct the error by typing through the keyboard.

Table 1 lists some statistics on PodCastle. The statistics were obtained between December 1st, 2006 (first release date), and January 26th, 2009. A total of 1489 episodes have been at least partially corrected. Nowadays, some episode registered in PodCastle are corrected almost everyday.

3.2. Podcast-dependent acoustic model training

In transcribing speech data from a specific task, a single or several acoustic models can be trained on the speech data matched with each task (task-dependent acoustic models). However, in the podcast-transcription task, it is not possible to train and optimize a very specific acoustic model because the acoustic conditions for each sub-task in podcasts vary widely.

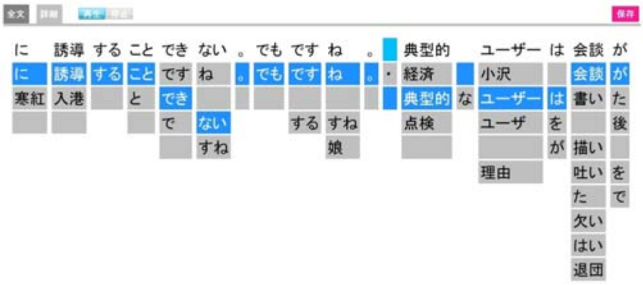


Figure 2: PodCastle screen snapshot of an interface for correcting speech recognition errors (competitive candidates are presented underneath the 1-best recognition results). Six errors in this excerpt were corrected by selecting from the candidates.

Therefore we require a method to cluster the variability of acoustic conditions, and to train the different acoustic models for each condition. In this work, as a first step, we introduce a method to train the acoustic models for each podcast with corrected transcriptions by PodCastle users (podcast-dependent acoustic models). It is beneficial to adopt this approach because the acoustic conditions in a podcast, such as speakers, recording conditions, and the level of background noise, tend to be similar across all the episodes. Furthermore, a podcast architecture (RSS) can be used such that the system can recognize the podcast to which an episode belongs, i.e., the appropriate acoustic model can be used in the recognizer for every episode (input mp3 audio file).

In the following subsections, we describe the process of podcast-dependent acoustic model training.

3.2.1. Audio segmentation

In order to deal with a continuous stream of audio, we conduct acoustic event detection and segmentation to obtain speech segments. In this work, we considered only three types of acoustic events: speech, music without speech, and other background sounds. The speech and background-sound models were trained on a Japanese speech corpus. For the music-without-speech model, we used the RWC Music Database (RWC-MDB-P-2001, RWC-MDB-R-2001, and RWC-MDB-J-2001) [11]. The entire audio stream is directly recognized using a conventional Viterbi decoder with the three GMMs in parallel. Since this approach tends to provide short segments of those events, we use an inter-class transition penalty that forces the decoding process to produce longer segments. Then, the transcription corrected by the users is divided into these speech segments according to the corresponding time intervals.

3.2.2. Obtaining phoneme transcription

The word transcription of each speech segment is converted to phoneme transcriptions by using pronunciation dictionaries. For the pronunciation dictionaries, we use a publicly available Japanese dictionary [12], and a Japanese Web service "*Hatena diary keyword*" [13] that publishes a list of new keywords including explanations and pronunciations. However, *unknown-pronunciation words* still remain in the error-correction results because podcasts cover a large range of topics in which the

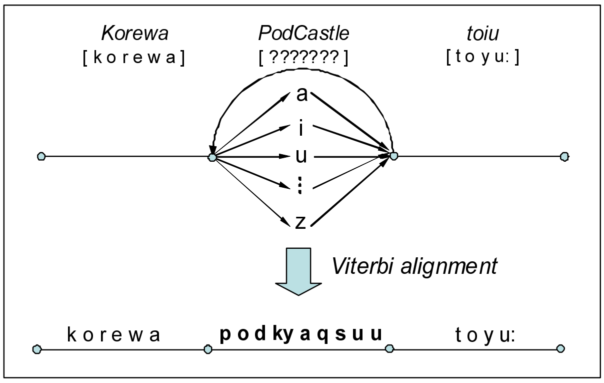


Figure 3: Viterbi alignment for obtaining phoneme transcription (the phoneme-loop sub-network for the unknown-pronunciation word “PodCastle” and the correct transcriptions for the other two words).

newest proper names and buzzwords appear. We therefore try to estimate the phoneme transcriptions (pronunciations) of these words from the actual speech data. As shown in Figure 3, we carry out the Viterbi alignment using the hybrid phonetic network that consists of phoneme-loop sub-networks for unknown-pronunciation words and exact phoneme sequences for other words.

3.2.3. Estimation of acoustic model parameters

The phoneme transcriptions obtained by the above process can be used to train the acoustic models for each podcast. The amount of speech data for training varies with each podcast and is not very large compared to that of task-dependent acoustic models. Hence, adaptation techniques such as MLLR [15] and MAP [16] are reasonable for estimating the parameters of podcast-dependent acoustic models. In this work, we apply the combination of MLLR and MAP, i.e., the MLLR transformed parameters are used as the priors for MAP estimation [17].

4. Experiments

To investigate the effectiveness of our method, we conducted recognition experiments using actual podcast speech data.

4.1. Podcast speech data

In this work, we used three Japanese podcasts and selected two episodes from each podcast as shown in Table 2. The test set includes three categories of podcasts: a daily news distributed by a Japanese broadcasting company (ID: A), a chitchat show by a Japanese popular singer (ID: B), and a quiz show on general knowledge hosted by a radio personality (ID: C).

For those podcasts, many of error parts were actually corrected by anonymous users through PodCastle. As the training data for each podcast, we used all the episodes which were corrected at least one part as shown in Table 3.

4.2. Baseline speech recognition system

For the podcast transcription system, a multi-pass decoding strategy is adopted. Recognition is performed in three decoding passes (stages) as follows:

1. Initial hypotheses are generated using a phoneme recognizer, and the hypotheses are used for unsupervised MLLR adaptation [15].

Table 2: Description of the test set.

ID	Category	Length (sec.)	Read or Spontaneous	Acoustic Conditions
A	news	2282.56	read	music
B	chitchat	2845.26	spontaneous	clean
C	quiz show	846.76	spontaneous	clean

Table 3: Description of the training set.

ID	# Episodes	Length (hours)
A	67	18.61
B	56	20.56
C	30	7.09

2. Word decoding is carried out using the adapted acoustic model. First, a word graph is generated using a lexical tree beam search with a 2-gram back-off language model. Then, the word graph is rescored using a 3-gram language model and the word hypotheses are used for the MLLR adaptation.
3. The above mentioned word decoding is conducted again using the adapted acoustic model, and the word graph is reconstructed. Finally, the consensus decoding [14] that directly minimizes the word error rate is conducted, and a confusion network is generated.

As a baseline acoustic model, a tied-state triphone HMM was trained with 600 h of presentation speeches from the Corpus of Spontaneous Japanese (CSJ) [18]. This triphone HMM was also used as the initial model for MLLR-MAP method in podcast-dependent acoustic training. For the language model, we used a large-scale N-gram model that includes as many words to be recognized as possible for podcast-transcription task [4]. A 3-gram language model was trained on the three kinds of text corpora/resources (newspapers, the CSJ, and web news) and contained 207,015 words.

4.3. Results

Table 4 summarizes the word error rates for each podcast. In this table, “w/o corrections” indicates that podcast-dependent acoustic model training was performed without error corrections, namely, fully unsupervised training with speech recognition results was performed, and “with corrections” indicates that podcast-dependent acoustic model training with error correction. All the experiments were carried out by using the recognition system described in Section 4.2 — i.e., the differences among the experiments are in the acoustic models used at Stage 1 of the three decoding passes.

The baseline results show that the recognition performance was not very high, regardless of applying the unsupervised MLLR adaptation in the recognition system. From the result of “w/o corrections”, the word error rate was reduced for all the podcasts. This suggests that the availability of sufficient training data for each podcast is important in acoustic model training even though the phoneme transcriptions are not perfect. With the error corrections by the users, the recognition performance

Table 4: Word error rate for each methods (%).

ID	baseline	w/o corrections	with corrections
A	16.88	15.12	**13.24**
B	30.98	25.15	**22.21**
C	35.16	30.67	**23.54**

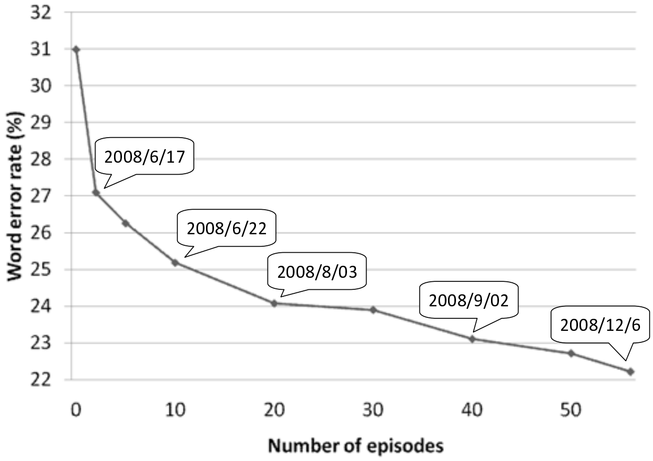


Figure 4: Word error rates for the number of episodes used in the acoustic model training. The dates on which the given number of corrected episodes were obtained are also shown.

was further improved for all the podcasts. In particular, the improvement for podcast C was relatively large. This is because the proposed method could reduce the acoustic mismatch between the baseline model and the distinctive manner of speaking (highly inflected speech) by the speaker in podcast C.

Finally, we investigated the effectiveness of the acoustic model training in terms of the number of episodes and how much improvement could be obtained in running the Web service. The focus in this experiment was podcast B, in which an episode is distributed once a week, and the recognition errors have been corrected in almost all the episodes. As can be seen in Figure 4, the largest improvement was obtained on June 17, 2008 (the number of episodes is 2), and subsequently, the performances gradually improved with an increase in the volume of training data. The recognition results of the acoustic model finally obtained in this experiment indicate that many words specific to podcast B (e.g., song name, place name, and radio program name) were included in the rest of the recognition errors. This suggests that podcast-dependent language modeling is needed for achieving further improvements and making better use of the users' contributions.

5. Conclusion

In this work, we have explored acoustic-model-training methods in podcast transcription task. To overcome the difficulties in a typical corpus-based approach, we introduced the wisdom of crowds, i.e., error corrections made by anonymous users were Incorporated into acoustic model training. Furthermore, we studied and developed a podcast-dependent acoustic model training system to deal with the diverse acoustic conditions in podcasts. The experimental results have shown that the system can significantly improve the recognition performance in podcast transcription. The system presented here is presently functioning as a part of PodCastle (URL: *http://podcastle.jp*).

In future work, we plan to further study other training techniques, particularly language modeling, for the speech recognition system so as to make best use of user corrections.

6. Acknowledgements

This work was partially supported by Grant-in-Aid for Scientific Research (B) (KAKENHI 19300065). We thank the anonymous users of PodCastle for correcting speech recognition errors.

7. References

[1] D. Miller, *et al.*: "BBN at TREC7: Using Hidden Markov Models for Information Retrieval", in *Proc. of TREC-7*, pp.133–142, 1999.

[2] J.-M. V. Thong, *et al.*: "Speechbot: An Experimental Speech-based Search Engine for Multimedia Content on the Web", *IEEE Transactions on Multimedia*, vol.4, no. 1, pp. 88–96, 2002.

[3] M. Goto, J. Ogata, and K. Eto: "PodCastle: A Web 2.0 Approach to Speech Recognition Research", In *Proc. of Interspeech 2007*, pp.2397–2400, 2007.

[4] J. Ogata, M. Goto, and K. Eto: "Automatic Transcription for a Web 2.0 Service to Search Podcasts", In *Proc. of Interspeech 2007*, pp.2617–2620, 2007.

[5] J. Mizuno, J. Ogata, and M. Goto: "A Similar Content Retrieval Method for Podcast Episodes", In *Proc. of SLT 2008*

[6] S. Renals, T. Hain, and H. Bourlard: "Recognition and Understanding of Meetings the AMI and AMIDA Projects" In *Proc. of ASRU 2007*, pp.238–247, 2007.

[7] J. Glass, *et al.*: "Recent Progress in the MIT Spoken Lecture Processing Project", In *Proc. of Interspeech 2007*, pp.2553–2556, 2007.

[8] L. Lamel, *et al.*, "The Limsi 2006 TC-STAR EPPS Transcription Systems", in *Proc. of ICASSP 2007*, pp.997–1000, 2007.

[9] ETSI ES 202 050 v1.1.1 STQ: "Distributed Speech Recognition; Advanced Front-end Feature Extraction Algorithm; Compression Algorithms". 2002.

[10] J. Ogata and M. Goto, "Speech Repair: Quick Error Correction Just by Using Selection Operation for Speech Input Interfaces", in *Proc. of Eurospeech 2005*, pp.133–136, 2005.

[11] M.Goto, "Development of the RWC Music Database," in *Proc. of ICA 2004*, pp.I-553–556, 2004.

[12] ipadic, http://chasen.naist.jp/stable/ipadic

[13] Hatena Diary Keyword, http://d.hatena.ne.jp/keyword

[14] L.Mangu, E.Brill and A.Stolcke: "Finding Consensus in Speech Recognition: Word Error Minimization and Other Applications of Confusion Network", Computer Speech and Language,Vol.14, No.4, pp.373-400, 2000.

[15] C.L.Leggetter and P.C.Woodland: "Maximum Likelihood Linear Regression for Speaker Adaptation of Continuous Density Hidden Markov Models", Computer Speech and Language,Vol.9, pp.171-185, 1995.

[16] J.L.Gauvain and C.H. Lee: "Maximum a Posteriori Estimation for Multivariate Gaussian Mixture Observations of Markov Chains", IEEE Trans. on Speech and Audio Processing, Vol.2, no.2, pp.291–298, 1994.

[17] E.Thelen, X.Aubert, P.Beyerlein: "Speaker Adaptation in the Philips System for Large Vocabulary Continuous Speech Recognition", in *Proc. of ICASSP'97*, pp.1035–1038, 1997.

[18] T.Kawahara, *et al.*: "Benchmark Test for Speech Recognition Using the Corpus of Spontaneous Japanese", in *Proc. ofSSPR 2003*, pp.135–138, 2003.