PodCastle: A Spoken Document Retrieval System for Podcasts and Its Performance Improvement by Anonymous User Contributions

Jun Ogata
National Institute of Advanced Industrial Science and Technology (AIST), JAPAN
jun.ogata@aist.go.jp

Masataka Goto
National Institute of Advanced Industrial Science and Technology (AIST), JAPAN
m.goto@aist.go.jp

ABSTRACT

We have developed a Web 2.0 service, *PodCastle*, that enables full-text searching of speech data (podcasts) on the basis of automatic speech recognition. PodCastle enables users to search and read podcasts, and to share the full text of speech recognition results for podcasts. However, even state-of-the-art speech recognizers cannot correctly transcribe podcasts, because podcasts' content and recording environments vary widely. PodCastle therefore encourages users to cooperate by correcting speech recognition errors so that podcasts can be searched more reliably. Furthermore, using the resulting corrections to train our speech recognizer provides a mechanism whereby the speech recognition performance is gradually improved. In our experiences from its practical use over the past 30 months (since December, 2006), we confirmed that the performance of PodCastle was improved by a number of anonymous user contributions.

Categories and Subject Descriptors

H.3.5 [**Information Storage and Retrieval**]: Online Information Services—*Web-based services*; I.2.7 [**Artificial Intelligence**]: Natural Language Processing —*Speech recognition and synthesis*

General Terms

Performance, Design, Experimentation

Keywords

Spoken Document Retrieval, Speech Recognition, Web 2.0

1. INTRODUCTION

Large amounts of speech data have become available on the web since podcasts, video sharing services like YouTube, and other digital archives have become popular and widespread. Spoken document retrieval technologies are therefore in great demand, especially for web services.

With a focus on Japanese podcasts, we launched a podcast search service called *PodCastle* [1, 5] in 2006 that allows anonymous users to find podcasts which include a search term, read full texts of their recognition results, and easily correct recognition errors. Even if a state-of-the-art speech recognizer is used to recognize podcasts on the web, a number of errors will naturally occur. A typical approach to deal with speech content that cannot be properly recognized is to create a speech corpus including such content and prepare correct transcripts to train speech recognizers. This approach, however, is impractical for PodCastle because advance preparation of a corpus covering diverse podcast content would be too costly and time consuming. PodCastle therefore encourages users to cooperate by correcting these errors. The resulting corrections can then be used not only to immediately improve the spoken document retrieval performance for corrected podcasts, but also to gradually improve the speech recognition performance by training our speech recognizer so that other podcasts can be searched more reliably. This approach can be described as *collaborative training for speech recognition.*

SSCS'09, October 23, 2009, Beijing, China.
ACM 978-1-60558-762-2/09/10.

2. OVERVIEW OF PODCASTLE

PodCastle allows full-text results of speech recognition to be accessed by both users and external search services, and allows a number of users to cooperate with each other to improve the speech recognition performance. Using these recognition results and RSS feeds, PodCastle has three main functions as follows.

2.1 Full-text searching function

PodCastle displays a list of podcast episodes[1] that include a search term typed in by a user. Each episode is accompanied by a text excerpt of speech recognition results around the highlighted search term and the user can listen to each excerpt. When the user clicks to select an episode, the system moves to the *reading function* described below.

2.2 Reading function

A user can "read" the full text of a podcast episode transcribed by our speech recognizer and grasp its content even without playing back the audio. Since PodCastle makes all the transcribed texts open to the public as *permalinks*, the PodCastle web page of a podcast episode can be found not only on PodCastle but also on various popular full-text search services.

2.3 Annotating function

A unique feature of PodCastle is that all users accessing the PodCastle service can annotate (i.e., transcribe) the full text of a podcast. However, transcribing manually from scratch is a costly and time consuming process for users. We therefore provide an efficient error correction interface earlier proposed [3]. As shown in Figure 1, a recognition result

[1] A podcast consists of several audio data (MP3 files) called *episodes* and a syndication feed (RSS) that includes metadata about the episodes.

Figure 1: The error correction interface in PodCastle (competitive candidates are presented underneath the normal recognition results). Five errors in this excerpt were corrected by selecting from the candidates.

excerpt is shown around the cursor and scrolled in synchronization with the audio playback. Each word in the excerpt is accompanied by other word candidates. Those candidates are generated beforehand by using a *confusion network* that is a simple network representing the intermediate recognition result [2]. This error correction by users can be regarded as a form of *social annotation* on Web 2.0. In PodCastle, the performance of both full-text search and speech recognition will be improved based on the annotation. Note that users are not expected to exhaustively correct all errors (although some users have done this, as described later), but they can be expected to casually correct some errors according to their interests.

2.4 Automatic Transcription of Podcasts

We had to overcome various difficulties to achieve our speech recognizer for podcasts. In terms of language modeling, for example, podcasts tend to include words and phrases related to recent topics, which are usually not registered in the system vocabulary. We therefore developed a method to keep a language model up-to-date by using on-line news texts [5]. In addition, in terms of acoustic modeling, podcasts include various types of speech data, such as pure speech, noisy speech, narrow-band speech, and speech with music. To reduce the acoustic mismatch, we apply several improvement methods such as noise suppression at the front-end and iterative unsupervised adaptation. The details of our speech recognizer are described in [5].

3. EXPERIENCES WITH PODCASTLE

PodCastle was released to the public at *http://podcastle.jp* on December 1st, 2006. So far, 554 podcasts have been registered, consisting of 48,781 episodes in total (as of June 12, 2009). Of these, 1795 episodes have been at least partially corrected. At present, some podcasts registered in PodCastle are corrected almost everyday or every week. We found that there are users who voluntarily cooperate in the correction, as happens with other Web 2.0 services, and that podcasts recorded by famous artists and TV personalities tend to receive many corrections.

For the collaborative training of our speech recognizer, we introduced a podcast-dependent acoustic model that is trained for each podcast using its transcripts corrected by anonymous users [4]. Through our experiments, we confirmed that the speech recognition performance for some podcasts that received many error corrections was actually improved by the acoustic model training (relative error reduction of 21-33%) [4] and the burden of error correction was reduced for those podcasts. Furthermore, we are currently studying and evaluating collaborative training of language models.

We have inferred some motivations for users correcting speech recognition errors on PodCastle, though we cannot directly ask since the users are anonymous. These motivations can be categorized as follows:

- **Error correction itself is enjoyable and interesting**
  Since the error correction interface is carefully designed to be useful and efficient, using it, especially for quick and accurate operations by proficient users, could be a form of fun somewhat like a video game.
- **Users want to contribute**
  Some users would often correct errors not only for their own convenience, but also to altruistically contribute to better speech recognition and retrieval.
- **Users want their podcasts to be correctly searched**
  The creators of a podcast (podcasters) would correct recognition errors in their own podcast so that it can be more accurately searched.
- **Users like the content and cannot tolerate the presence of recognition errors in it**
  Some fans of famous artists or TV personalities would correct errors because they like the podcasters' voices and cannot tolerate the presence of recognition errors in their favorite content. In fact, we have observed that such podcasts generally receive more corrections than other types.

4. CONCLUSION

We have described PodCastle, a spoken document retrieval system that is continually improved by user contributions. In the future, we plan to support English podcasts because the underlying ideas of PodCastle are universal and language-independent. This study has aimed at enabling *collaborative training for speech recognition*, where full-text transcripts containing recognition errors are first disclosed and then corrected by anonymous users. In practice, such training has significantly improved the recognition performance in podcast transcription. To refer to this research approach, we introduced and defined the term *Speech Recognition Research 2.0* [1] that brings the benefits of Web 2.0 to speech recognition research. We hope that this study will demonstrate the importance and potential of incorporating user contributions into speech recognition, and that various other projects following the Speech Recognition Research 2.0 approach will be done, thus adding a new dimension to this field of research.

5. REFERENCES

[1] M. Goto, J. Ogata, and K. Eto. PodCastle: A Web 2.0 approach to speech recognition research. In *Proc. of Interspeech 2007*, pages 2397–2400, 2007.

[2] L. Mangu, E. Brill, and A. Stolcke. Finding consensus in speech recognition: Word error minimization and other applications of confusion networks. *Computer Speech and Language*, 14(4):373–400, 2000.

[3] J. Ogata and M. Goto. Speech Repair: Quick error correction just by using selection operation for speech input interfaces. In *Proc. of Eurospeech 2005*, pages 133–136, 2005.

[4] J. Ogata and M. Goto. PodCastle: Collaborative training of acoustic models on the basis of wisdom of crowds for podcast transcription. In *Proc. of Interspeech 2009*, 2009.

[5] J. Ogata, M. Goto, and K. Eto. Automatic transcription for a web 2.0 service to search podcasts. In *Proc. of Interspeech 2007*, pages 2617–2620, 2007.