Podcast Archives: Access Through SpeechIndexer Technology

Ulrike Glavitsch (1), Dennis Küpper (1), Tobias Stamm (2), Jozsef Szakos (3)
(1) EMPA, Swiss Federal Laboratories for Materials Science and Technology, Ueberlandstrasse 129, 8600 Dübendorf, Switzerland
(2) Manderim GmbH, c/o Tobias Stamm, Kornhausstrasse 19, 8037 Zurich, Switzerland
(3) The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong

Abstract

This paper presents an indexing and retrieval method for podcasts where transcripts exist. The indexing units are speech segments limited by speech pauses. Podcasts and their corresponding texts are synchronized automatically in terms of these indexing units by the software SpeechIndexer. A web-based interactive player function makes each podcast fully accessible. This is demonstrated with PodClub – a podcast service for language learning offered by the largest language school in Switzerland. The interactive player received enthusiastic feedback in a pilot test phase and is online since January 2014. The search across podcast archives is performed with SpeechConcordancer, which belongs to the SpeechIndexer software suite.

Introduction

Radio and TV stations maintain large archives of podcasts. They grow continuously and are kept for a long time. However, access and retrieval is difficult. Current technology, e.g. Windows Media Player, only allows access to the audio file by means of absolute or relative time positions. Content-based access is rarely possible since state-of-the-art automatic speech recognition (ASR) technology is neither efficient nor reliable enough to transcribe arbitrary speech files independently of the speaker. As a consequence, most radio and TV stations use multimedia databases where the podcasts together with corresponding meta information, e.g. author, keywords, etc. are stored. The Swiss radio and television corporation uses the media database FARO for archiving TV broadcasts and will migrate their radio archive to FARO by the end of 2014 [1].

Some stations, however, publish podcasts together with the transcripts. An example is the weekly episode "In Touch" on BBC Radio 4 where the transcript is posted after one or two weeks [2]. Audio and text files are presented as two separate entities, i.e. there are no links or cross-references between the two. The TED talks offer the possibility of showing an interactive transcript that visualizes the currently spoken entity by underlining it [3]. The units of synchronizing audio and text file are a sequence of several words. However, starting the video at any position in the text and navigating forward and backward is not realized so far.

Current search engines, e.g. Google, Yahoo, Lycos, perform a content-based retrieval on HTML texts and present the results as a ranked list of pages [4]. Boolean search engines are common as well, such an engine is used by the British National Corpus that contains a 100 million word collection of written and spoken text [5].

In this paper, we present SpeechIndexer technology as an approach to access individual podcasts and podcast archives. The use case is PodClub – a service by the club schools Migros, which is the largest language school in Switzerland [6]. Migros publishes podcasts and the corresponding transcripts in five different languages every other week since 2008. SpeechIndexer is a stand-alone software for describing audio files segment-wise and correlating the textual descriptions with the corresponding speech segments. In case a transcript is given SpeechIndexer aligns the audio and text files in units of speech segments limited by speech pauses [7, 8, 9]. As a result, the aligned text is marked and the user listens to a speech segment by following the marked text part. The link between a speech segment and its corresponding text part is called index. Manual indexing is supported by a built-in pause finder that automatically segments a speech file into pause and speech segments [10]. For the case at hand, SpeechIndexer was extended by an automatic alignment function to synchronize audio and text files in much less time. The output files of the automatic alignment are used to control a web-based interactive player that allows for accessing the podcast at an arbitrary position. The text part of the currently played speech segment is highlighted. Furthermore, the user may navigate through the text by jumping to the next or previous index. User feedback on a pilot test version of the interactive player was encouraging so that the function went online at the beginning of 2014. Furthermore, we introduce the SpeechConcordancer program as part of the SpeechIndexer software package to search across archives of indexed podcasts. SpeechConcordancer performs a Boolean search and lists all search results with the enclosing left and right context. SpeechConcordancer was originally developed for corpus-based language learning but is equally well-suited for speech document retrieval.

The outline of the paper is as follows. The first section describes the offline synchronization of the podcast audio and text files with SpeechIndexer. We report on the web-based interactive player function and its design in the next section. The third section gives an overview on the user feedback of the pilot test. The fourth section presents the program SpeechConcordancer for searching across podcast archives. Finally, we draw conclusions and give an outlook to future work.

Synchronization of podcast and transcript

The automatic synchronization of podcasts with the texts is performed offline using an extended version of SpeechIndexer. The whole process of synchronizing the podcasts occurs after the podcast text and mp3 file are loaded to the web page, but before it is made publicly visible. This way, we can generate a text file in UTF-8 format from the authoritative HTML page to guarantee maximum accordance between the uploaded text and the text used in SpeechIndexer.

Several steps are required to align the audio and text files automatically and the whole process had to be user-friendly since it is executed by people that were not directly involved in the development, i.e. Migros personnel. Thus, we developed an easily operated graphical user interface (GUI) that visualizes the result of each step.

The audio and text files are not aligned directly. Each podcast contains a music introductory part at the beginning, some short sounds the middle and final notes at the end. These music intermezzos are equal for each podcast series and have to be excluded from the synchronization. Currently, there are six different podcast series in five different languages. There are two podcast series for English for different language levels. In addition, the podcast text contains portions that are not spoken. For instance, each podcasts has a headline with the series title, some keywords and the issue date. The first step consists of recognizing all those parts that are excluded from the synchronization. The main step is automatic alignment of the podcast with the text, which is performed using an automatic speech recognition component.

Recognition of podcast series and music intermezzos

The podcast series is recognized by inspecting the configuration file. The configuration file describes each podcast series with information about the series title, language, media files for the music intermezzos, dictionary file and individual parameters for the automatic alignment. The configuration file is read at SpeechIndexer start. When loading a text we check whether one of the podcast series title occurs in the podcast head line. The series where the title matches is stored internally as the current podcast series.

The algorithm to find the music intermezzos uses the media files of the current podcast series as input parameters. The media files contain the introduction music, the middle piece and the final notes and are referred to as patterns in the following. Our algorithm first downsamples the audio and pattern files to 1000 Hz to speed up the process. This resolution is still sufficient for our purposes. It then computes the unnormalized cross correlation between the main audio file and the patterns. This can be done in Fourier space, which is significantly faster. The normalization is not needed because the music intermezzos are always played at the same volume and are about equally loud as the surrounding spoken text. The recognition of music intermezzos for a podcast of roughly 15 minutes takes less than 10 seconds. This is acceptable given that a progress bar informs the user about the advance of the recognition. Recognized music segments are visualized in SpeechIndexer by a different color. The user may correct the output of the algorithm manually by deleting and inserting regions to be excluded.

Automatic synchronization

The automatic synchronization of a podcast with its text uses a commercial automatic speech recognition (ASR) component for the word alignments and an algorithm to build the alignment units. As mentioned before, the alignment units are speech segments limited by pauses, i.e. spoken in one breath of air.

Several commercial ASR systems were evaluated before project start. We decided in favor of Annosoft's Text BasedLipsync Software Development Kit (SDK). Text Based

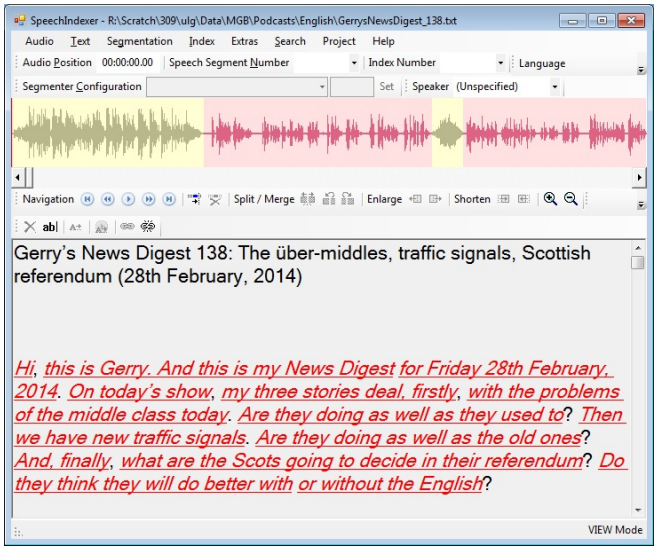

***Figure 1**. SpeechIndexer main window with loaded speech file and marked text. The excluded segments in the signal window are shown in lighter color.*

Lipsync by Annosoft offers text-to-speech alignment for all podcast languages and provides mechanisms for excluded audio segments as they occur in our case.

The output of the text-to-speech alignment is both a list of phoneme matches and a list of word matches. A match consists of the phoneme or word and its corresponding sample start and end positions in the audio file. Annosoft's Text Based Lipsync SDK was integrated into SpeechIndexer and its commands are called from the graphical user interface (GUI).

The algorithm to build the alignment units from the word alignment output uses two parameters: (1) the minimum pause length and (2) the maximum alignment unit length. Both parameters are specified for each podcast series individually. They depend heavily on the speaking rate. These values were set empirically. For instance, the lowest minimum pause length of 250 ms was found for the French podcasts and the English podcasts of language levels B2 to C2. The highest minimum pause length of 350 ms resulted for the German podcast series and the English podcasts for levels A2 to B1. The maximum alignment lengths were set to values between 6 and 8 seconds. Our algorithm searches the list of word matches for pauses between words larger than the configured minimum pause length. An indexing unit is the speech segment between the two pauses. The speech segment is split if it exceeds the maximum alignment length. The split position is the first speech pause that is at least half as long as the configured value. The speech segment remains undivided if no such split position is found. The result of the automatic synchronization is visualized in SpeechIndexer. Each alignment unit is written in red, italics and is underlined. Fig. 1 shows the outcome of the synchronization step in the main window of SpeechIndexer. The excluded music segments are visible in a lighter color in the upper signal window and the indices appear as

marked text in the text window. The head line with the title and the date of issue is not marked. It is not spoken and automatically excluded from the synchronization. For a podcast of 15 minutes length the synchronization with the text in terms of these indexing units takes roughly 20 seconds. Notification about the progress of the automatic synchronization is shown by a graphical element.

The user may check the rough correctness of this step by following some of the links and listen to what is spoken. Then, he saves the indices to a file. The index file is in XML format and contains for each index the sample start and end position of the speech segment as well as start and end position of the corresponding text part. Both the text file and the index file are loaded to the server of PodClub.

Web-based interactive player

A web program controls the interactive player for the podcasts using the output of the automatic synchronization. The interactive player highlights every text segment currently spoken and allows playing from an arbitrary position in the text. The highlighted text segments correspond to the units found during the automatic synchronization of the audio and text file. In addition, users may navigate on the text, i.e. they can jump to the previous or next index. Links to the glossary remain active.

The web program is written in Javascript. It uses both the index file and the UTF-8 text file. The program receives the currently played position in the audio file periodically. For each such position it inspects the list of indices for a matching index, i.e. an index where the given sample position is between the start and end position of the speech segment. The text part is found from the UTF-8 text file using the text start and end positions of the index. Now the same text part is to be found in the uploaded HTML text which is performed with a grabber function. The grabber scans the HTML text for text strings that match the given text part. The grabber accounts for the different coding of characters in HTML and UTF-8 documents. As soon as such a match is found the text part in the HTML page is highlighted. The highlighting of the text part is cleared as soon as the current sample position of the audio is not within the current index any longer. So far, the interactive player works on all browsers on PCs, Macs, iPads and iPhones.

The player elements such as start and stop buttons were carefully designed and a few additional functions for user-friendliness were added. For instance, the user may select the highlighting color from a set of given ones, he may disable the highlighting of indices and he may vary the font size.

Pilot test phase

The club schools Migros performed a pilot test phase with a single indexed podcast online and asked for user feedback. The pilot test phase took place in September 2013 and more than 1000 persons (language school teachers, learners and PodClub users) were asked to fill out the survey. Migros received 39 answers, 35 from the German part of Switzerland and 4 from the French part. The given feedback was very positive. None of the persons reported technical problems and they were able to run the new function on various browsers and devices. The majority of the people liked the new interactive player function a lot and was praising it. They said that it helps them to learn the language and that they can repeat text parts they did not understand. Moreover, they reported that they better follow the podcast this way and like the lengths and coloring of the indices. Two people said that they do not need the new function since they already know the language well and just want to listen to the podcast. Among additional comments a few users would like an online lexicon in connection with the text that shows the translations of words like in iBooks.

After such encouraging comments the club schools Migros redesigned the graphical elements for the player, selected the colors for highlighting and entered the options for different font sizes. The first series of podcasts with the interactive player went online on January 31, 2014. Since then, podcasts with the interactive player function have been published every other week and the new function finds approval in user comments.

Searching podcast archives

The SpeechIndexer software suite contains a component called SpeechConcordancer that performs searches across archives of indexed speech documents. This offline program may be used to search the podcast archives of the PodClub – not web-based but behind the scenes - as is shown in the following.

SpeechConcordancer comes with a graphical user interface (GUI) where the user enters the list of synchronized speech files. These files are given as so-called project files that contain the path names of the involved files, i.e. audio, text and index files. Project files are generated easily within the main SpeechIndexer. The second input is the search term. It is selected from a word list generated from the podcast texts. The output is a list of so-called concordances where the search term is written in the middle surrounded by the left and right context whose length can be specified. The index of the search term is displayed as in SpeechIndexer, i.e. red, italics and underlined. Users may listen to each search result, i.e. the index, while they see the name of the podcast in the list of project files highlighted. Language learners may study the pronunciation variants of a given word by listening to the speech segments in which the word occurs and they learn the various contexts of a word. Persons interested finding passages of a podcast of a given topic may open the podcast indicated by the filename in the project list.

Fig. 2 shows the result of a search with SpeechConcordancer. The sample archive contains issue 129 to 138 of the English podcast series “Gerry’s News Digest” and the search term is “scotland”. The search results are numbered (1) to (8). Fig. 2 shows the selection of the first search result and the selection of the podcast in which the term occurs, namely issue 136.

Conclusions and Outlook

We presented an indexing and retrieval method for podcasts by means of SpeechIndexer technology. The indexing units are speech segments spoken in one breath of air and limited by speech pauses. An indexed podcast is fully accessible – either by pointing to a text position or by using the built-in search function of the browser. The SpeechConcordancer program serves as Boolean retrieval engine as it finds all podcasts that contain a given search term.

The development of the web-based interactive player was an engineering task of high complexity. The integration of commercial software in SpeechIndexer, the development of algorithms as well as the design and implementation of the user

control were not easy. However, the encouraging feedback from both Podclub users and Migros technical staff showed that both

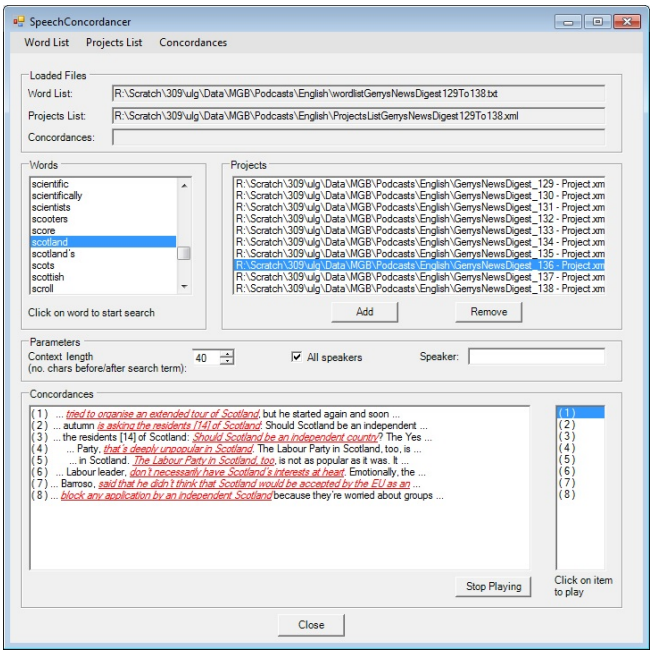

Figure 2. Search of podcast archive for term "scotland". Search results are shown as concordance list where each item can be listened to individually.

programs, i.e. the interactive player and the extended version of SpeechIndexer, are well accepted.

It has to be emphasized that our SpeechIndexer technology is not limited to accessing podcasts where a transcript is given. As mentioned in the introduction, descriptions for audio files may be entered manually for speech sections of arbitrary length. These descriptions are linked with the corresponding speech segment. This way, radio news podcasts may, for instance, be indexed by topic where the topic keywords are associated with the corresponding audio section. A search for a topic in SpeechConcordancer will find the podcasts that deal with the topic and at the same time make the appropriate speech section immediately accessible. Thus, SpeechIndexer turns out to be a very flexible indexing tool for various types of podcasts.

Future work will concentrate on developing our own automatic alignment software in order to become independent of a commercial tool.

Acknowledgements

The development of the automatic synchronization of podcasts with their texts in units delimited by speech pauses was supported by a grant from the Commission for Innovation and Technology (CTI) of the Swiss Confederation.

References

[1] FARO: Neue digital Archivplattform für Schweizer Fernsehen, www-05.ibm.com/de/media/news/faro-18-04-07.html (2007)

[2] BBC Radio 4 In Touch, www.bbc.co.uk/programmes/b006qxww

[3] TED: Ideas worth spreading, www.ted.com

[4] S. Brin, L. Page: The Anatomy of a Large-Scale Hypertextual Web Search Engine, In: Computer Networks and ISDN Systems, 30:107-117. (1998).

[5] British National Corpus, www.natcorp.ox.ac.uk

[6] Klubschule Migros PodClub, www.podclub.ch

[7] J. Szakos, U. Glavitsch, Seamless Speech Indexing and Retrieval: Developing a New Technology for the Documentation and Teaching of Endangered Formosan Aboriginal Languages, Proc. Int'l Conference on Education and Information Systems: Technologies and Applications (EISTA'04), pg. 88-93. (2004).

[8] J. Szakos, U. Glavitsch, SpeechIndexer in Action: Managing Endangered Formosan Languages, Proc. Interspeech, pg. 4017-4019. (2007).

[9] U. Glavitsch, K. Simon, J. Szakos, SpeechIndexer: A Flexible Software for Audio-Visual Language Learning. Proc. Int'l Conference on Education, Informatics and Cybernetics (icEIC), pg. 79-82. (2011).

[10] L. R. Rabiner, M. R. Sambur, An Algorithm for Determining the Endpoints of Isolated Utterances, Bell System Technical Journal, 54:297-315. (1975).

Author Biography

Ulrike Glavitsch studied Computer Science at ETH Zurich (Dipl. Ing., 1988) and received a Master's degree from Stanford University, USA, in 1990. She worked as software developer for air traffic control systems for some years before returning to academia. Since 2010 she is a scientific collaborator at the EMPA Media Technology Lab. Her research interests include automatic speech recognition, speech indexing and retrieval as well as language learning software.

Dennis Küpper joined the EMPA Media Technology Lab in January 2009 as a research associate after receiving his master's degree in Computer Graphics from ETH Zurich. His research interests include gamut and tone mapping, color spaces, HDR capture and display, psychovisual testing, computer graphics, audio processing, low-level programming, and efficient algorithms and data structures in general.

Tobias Stamm received his Master in Computer Science from ETH Zurich in 2007 specializing in computer graphics. He developed a complete color workflow based on frequency modulated halftoning (patented), incorporating gamut mapping, separation and dot gain compensation. Furthermore, he provided the implementation of the light scattering simulation Scatter3D. He founded his own company, Manderim GmbH, in 2011 which provides software engineering services and expertise in research and development.

Jozsef Szakos is Associate Professor at the Chinese and Bilingual Studies Department, Hong Kong Polytechnic University. He worked in Taiwan, as director of the Indigenous Languages Department, National Dong Hua University, doing speech corpus based documentation on Formosan languages. He earned his Dr Phil in Bonn, Germany (1994) with a dissertation on Tsou, an Austronesian language of Taiwan, majoring in General Linguistics, Sinology and Comparative Religion.