Pattern Recognition Letters 71 (2016) 8–15

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Assisted keyword indexing for lecture videos using unsupervised keyword spotting☆

Manish Kanadje[a,*], Zachary Miller[a], Anurag Agarwal[b], Roger Gaborski[a], Richard Zanibbi[a], Stephanie Ludi[c]

[a] Department of Computer Science, Rochester Institute of Technology, 102 Lomb Memorial Drive, Rochester, NY 14623-5608, United States
[b] School of Mathematical Sciences, Rochester Institute of Technology, 102 Lomb Memorial Drive, Rochester, NY 14623-5608, United States
[c] Software Engineering Department, Rochester Institute of Technology, 102 Lomb Memorial Drive, Rochester, NY 14623-5608, United States

ARTICLE INFO

Article history:
Received 21 July 2015
Available online 2 December 2015

Keywords:
Keyword spotting
Unsupervised
Lecture indexing
Mel Frequency Cepstral Coefficients
Segmental Dynamic Time Warping
Within-speaker query

ABSTRACT

Many students use videos to supplement learning outside the classroom. This is particularly important for students with challenged visual capacities, for whom seeing the board during lecture is difficult. For these students, we believe that recording the lectures they attend and providing effective video indexing and search tools will make it easier for them to learn course subject matter at their own pace. As a first step in this direction, we seek to help instructors create an index for their lecture videos using audio keyword search, with queries recorded by the instructor on their laptop and/or created from video excerpts. For this we have created an unsupervised within-speaker keyword spotting system. We represent audio data using de-noised, whitened and scale-normalized Mel Frequency Cepstral Coefficient (MFCC) features, and locate queries using Segmental Dynamic Time Warping (SDTW) of feature sequences. Our system is evaluated using introductory Linear Algebra lectures from instructors with different accents at two U.S. universities. For lectures produced using a video camera at RIT, laptop-recorded queries obtain an average Precision at 10 of 71.5%, while 79.5% is obtained for within-lecture queries. For lectures recorded using a lapel microphone at MIT, using a similar keyword set we obtain a much higher average Precision at 10 of 89.5%. Our results suggest that our system is robust to changes in environment, speaker and recording setup.

1. Introduction

In recent times, there has been a significant increase in digital content in order to supplement the learning of students. Video recordings of classroom lectures can help students to improve their understanding significantly. With video recordings, students may access lecture content multiple times according to their need. However, video lectures do not have a well-defined index. Students have to manually search to reach a point of interest. This is a tedious task. However, this task becomes increasingly difficult for people having challenges in visual capacities. A text-like index for the video content will be immensely helpful for such students, in order to improve the accessibility of video lectures.

AccessMath is a video lecture indexing and retrieval system being designed at our institute. The main goal of AccessMath system is to facilitate the learning of linear algebra lectures for students having challenges in visual capabilities. This paper describes the audio indexing portion of AccessMath. We plan for AccessMath to eventually be a lecture indexing and retrieval system accepting queries issued in image, audio or text formats. Using this system, a student could search a linear algebra lecture for a formula, e.g. $A\bar{x} = \bar{b}$, by selecting a part of an image or a spoken query from the lecture.

We propose a keyword spotting system which will enable an instructor or student to perform search using audio queries spoken by the instructor. We have also created a prototype to help instructors and students organize search hits generated by the system. This system helps create an index similar to the table of contents for a textbook using within-speaker audio queries. Keyword spotting is a relatively difficult task as differences in speech characteristics such as accent, pitch and environment cause high variance in utterances of the same keywords. In the proposed system, we have considered single channel audio input created in a single speaker environment.

Keyword spotting systems convert an input speech signal into a temporal spectral vector. After modeling the speech signal, these systems usually fall within two different categories: Dynamic Time Warping-based or Hidden Markov Model-based. Dynamic Time Warping (DTW) finds an optimal alignment between two audio sequences, seeking to determine whether they represent the same

☆ This paper has been recommended for acceptance by Nappi Michele.

* Corresponding author. Tel.: +91 9673043835.
E-mail address: manishkanadje@gmail.com (M. Kanadje).

http://dx.doi.org/10.1016/j.patrec.2015.11.012
0167-8655/

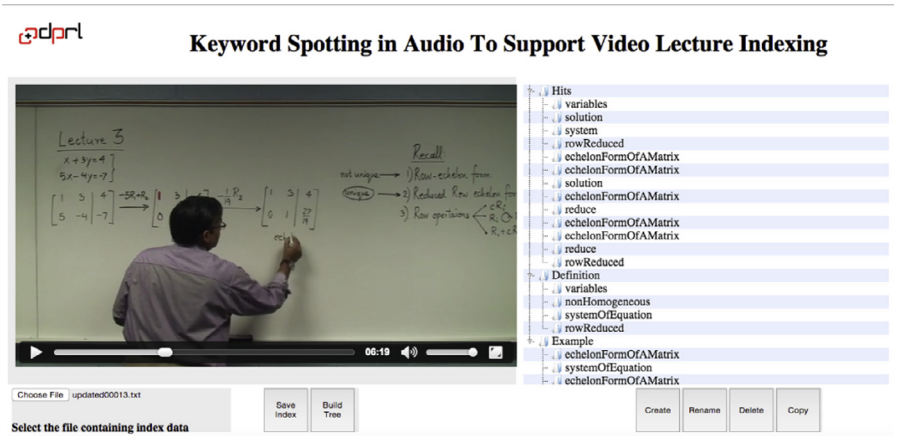

Fig. 1. Using indexing tools the user can play the video lecture from the point of generated hits. The tree based indexing structure helps the user to organize hits into groups such as 'Definition' or 'Example'.

word [16]. DTW matches two temporal sequences by non-linearly comparing audio frames and calculating the cost of alignment. In contrast, Hidden Markov Model-based approaches require training data for creating probabilistic temporal models for individual words. DTW does not require labeled data for training. However, the cost of computation is high for DTW, $O(mn)$ where m and n are sequence lengths. For this reason, many variations of DTW attempt to reduce its computational cost.

As shown in Fig. 1, our system creates an index of candidates for a query within a lecture.[1] We have offered the functionality of hierarchical annotations to make this index more useful. For example, it would be helpful if a user can create categories such as 'Definition' and 'Example' to organize query results, such as shown in Fig. 1. Once these categories are created, the user can drag and drop hits into categories. The user can also create copies of a search result, and then place it in multiple categories. Finally, the current index state can be saved in JSON format, and then later loaded to generate the same tree structure again.[2]

Our approach employs Mel Frequency Cepstral Coefficients [6] and a variation of Dynamic Time Warping algorithm called Segmental Dynamic Time Warping [15]. We have evaluated our system using videos from introductory Linear Algebra courses recorded at two different U.S. institutions (RIT and MIT). At RIT, a set of linear algebra lectures was recorded using a lone video camera in a classroom without students by one of the authors (Dr. A. Agarwal). Using queries recorded on a laptop by the instructor, our system achieved a Precision at 10 of 71.5%. Using the same queries extracted from the lecture audio, a Precision at 10 of 79.5% was obtained. The MIT lectures were recorded by an instructor with a different accent who used a lapel microphone for recording. Without modifying system parameters and using keywords similar to that used for the RIT lectures we obtained a much higher Precision at 10 of 89.5%, suggesting that our system is robust to different speakers and recording environments.

In the remainder of this paper, we summarize related work in Section 2, our keyword spotting methodology in Section 3, the experimental design and results in Sections 4 and 5, and then conclude and identify future directions in Section 6.

2. Related work

Previous systems have been proposed for indexing, retrieving and annotating video content. For example, the MIT Lecture Browser by Glass et al. [8] allows users to search lecture audio using text queries. Automatic speech recognition is used to create a transcript of the lecture audio, which can then be searched textually. This transcription-based index may not have temporal information, and may contain recognition errors for rarer terms outside the language model. Similar to the MIT Lecture Browser, the Speech@FIT Lecture Browser [18] uses speech recognition to support text search of lecture audio. This system shares many of the strengths and weaknesses of the MIT Lecture Browser. It also detects lecture slide changes using image features to provide pointers for lecture navigation.

The Video Audio Structure Text Multimedia (VAST MM) Browser designed by Haubold and Kender [10] is another example of an indexing and annotation system designed for video presentations. This system creates a visual index for speaker segmentation using changes in activities. It also offers textual indices for searching through the transcription of the video.

NTU Virtual Instructor [12] offers sophisticated tools for finding lecture recordings of interest, including automatic summarization and keyword detection. Keywords are linked to particular points in the lecture in which they occur, allowing the user to rapidly find relevant content. Bilingual automatic speech recognition is integral to the approach, which also supports text-based search of spoken terms.

While these systems support lecture annotation and textual search, they do not offer video search using audio queries. In our work we seek to support audio queries, and avoid the need to train speech recognizers for new lecturers. To do this, we have chosen to use unsupervised keyword spotting in audio.

Mel Frequency Cepstral Coefficients (MFCC) are frequently used to represent speech audio in keyword-spotting systems. MFCC features were first discussed by Davis and Mermelstein [6]. MFCCs are computed based on a model of how human ears perceive speech, and compensate for insignificant variations present in higher frequency bands. MFCC feature extraction is usually followed by normalization to reduce the impact of environmental mismatch. Alam et al. [1] have discussed different normalization approaches for MFCC features. The short-term mean variance approach is similar to the whitening process used in this paper. However, they have used *mean* (μ) and *standard deviation* (σ) values computed over a moving window instead of the complete sequence as done in this paper. Using parameters obtained from the complete sequence reduces processing time, which is important for a real-time system.

Noise reduction is used to remove non-speaker audio elements. The authors in [7,11] have discussed Cepstral Subtraction from the MFCC features for noise reduction. These techniques model slowly changing noise using a filtering approach. A noise profile is computed by filtering the input audio asymmetrically if the current intensity

[1] A working demo of the interface is available at https://www.cs.rit.edu/~dprl/keywords/index.html

[2] This prototype is created using 'jsTree' http://www.jstree.com/