Presentation Tracking Using Confusion Networks, Semantic Matching, and Keyword Weighting

Reza Asadi

asadi@ccs.neu.edu

August 2016

Abstract

Oral presentations are an essential yet challenging aspect of academic and professional life. To date, many commercial and research products have been developed to provide support for the authoring, rehearsal and delivery of presentations. however, little work has been conducted to provide real-time tracking of presentation content. Given the presentation slides with speaking notes, a presentation tracking system uses automatic speech recognition to track content coverage by the speaker. This can help speakers ensure that they cover their planned content while potentially reducing their speech anxiety and enabling various real-time presentation support technologies, such as automatic slide advance. Presentation tracking is, however, a complex task; due to the inaccuracy of current speech recognition systems and the fact that speakers rarely follow the exact presentation notes.

In this thesis, I present a novel framework for both on-line tracking of presentations at the sub-slide level, as well as global presentation tracking through a slide deck that allows for more speaker flexibility in choosing slides to present. Tracking is performed by semantic matching of the confusion network results from an automatic speech recognition system against the slide's content keywords. The keywords are selected and weighted based on word specificity and semantic similarity measures. My evaluation studies show that using confusion networks results in a more robust speech recognition system, while semantic matching reduces the reliance on the exact notes, and keyword weighting improves the accuracy of the tracking system. I will present my plans for improving tracking accuracy and addition of slide tracking to support more dynamic presentations. I plan to integrate this presentation tracking framework into two different applications to provide support for both presentation rehearsal and delivery, and conduct user studies to evaluate its effectiveness.

1. Introduction

Presentations are necessary but stressful tasks for almost everyone. The quality of a presentation can affect the speaker's professional and academic performance. Nowadays many software products are available to help improve the quality of presentations, ranging from commercial products that can help users create their presentations [1, 2] to systems like PitchPerfect [3], PresentMate [4], or Cicero [5] which aim to aid in the rehearsal process. There are also studies on using virtual characters instead of human speakers for presentation or co-presentation [6, 7]. Despite these advances, the average quality of professional presentations is still low [8] and further research is required.

One of the less explored areas of public speech assistance is presentation tracking. A presentation tracking system uses speech recognition to track the presentation content coverage by the speaker. Applications equipped with presentation tracking can provide better content-based assistance during presentation

rehearsal and delivery. The tracking information could be used to provide intelligent teleprompters that automatically highlight key phrases to remind the presenter what to say. Alternatively, the framework could also be used to develop intelligent virtual co-presenter systems, in which a virtual agent could track the presentation progress and automatically deliver parts that have been forgotten by the human presenter.

Presentation tracking is not a trivial task. In previous studies on presentation alignment and tracking, ASR was used to transcribe the presentations, and text alignment methods were used to match the script and the transcriptions [9, 10, 11]. These systems depend on the accuracy of the automatic speech recognition systems (ASR) which are not perfect yet [12]. Even having a perfect speech transcription does not result in perfect tracking because speakers rarely follow their presentation notes exactly [9, 10]. Therefore, the system cannot depend on the exact forms of sentences uttered. The input for an on-line tracking system is a speech audio stream which makes it different from off-line problems such as alignment of recorded speech to a transcription. This results in lack of knowledge about the overall structure of the spoken content that could otherwise help in associating utterance segments with slide content [11].

I propose a range of solutions to address these complications. One approach to dealing with imperfect ASR is to reduce the vulnerability to ASR errors. To do so, instead of using only the best hypothesis from ASR system, I process multiple, ambiguous hypotheses in the form of an ASR output graph called a confusion network [13]. Confusion networks provide powerful word representations and result in more robust ASR systems [14]. In order to reduce the reliance on exact notes, keywords and query expansion techniques can be used which can help in detecting the semantic relatedness between the spoken terms and the source text [15]. I extract the keywords from text and then match the ASR results with those keywords. Each time a keyword from a text segment is spotted in speech, the probability of that segment being covered is increased. The amount of increase depends on the discriminatory power of the keyword. In information retrieval studies, methods such as term frequency- inverse document frequency (*tf.idf*) have been used to extract and assign weights to the keywords based on their specificity in text [16].

I propose that using the above mentioned techniques I can increase the accuracy of presentation tracking. In this thesis, I present a framework for on-line presentation tracking at the sub-slide level based on semantic matching, confusion networks, and keyword weighting. Using this framework, I tackle the following research questions:

1. Will semantic matching reduce the reliance on exact notes?
2. Can confusion networks lower the dependency of tracking on ASR accuracy?
3. Does keyword weighting improve the accuracy of the presentation tracking?
4. How effective is tracking in presentation assistance applications?

To answer these questions, in a pilot study, I investigated different approaches for presentation tracking using the proposed framework. This study showed that semantic matching, confusion networks and keyword weighting can improve the accuracy of the tracking system.

I plan to improve the tracking accuracy and extend the framework to provide slide tracking in addition to sub-slide level tracking. I will further evaluate the effectiveness of this tracking framework by integrating it in two different speech assistance applications: A presentation rehearsal system and a co-presenter system [6]. The rehearsal system will measure the speech quality and, based on tracking results, provides sub-slide level quality feedback to the user. I plan to compare the effectiveness of providing feedback on slide segments compared to the entire slide or presentations. In the co-presenter study, I plan to use the tracking framework to provide dynamic turn-taking between a virtual agent and the speaker. The agent will be able to automatically deliver parts that have not been covered by the human presenter. I propose that using this system will result in less anxiety for the speaker and will cover more content.

In the following sections, I review the previous related work, present my proposed framework, and demonstrate the results of my evaluation study. I will then explain my planned work for further improvement and evaluation of the tracking framework.

2. Theoretical Background

In this section, I present a brief survey of currently available presentation assistance systems and the limited studies related to presentation tracking. I will then review the previous work on two main challenges in presentation tracking: speech recognition and semantic text retrieval.

2.1 Presentation Assistance Technologies

There are several aspects of a presentation that can have an effect on its quality: the content of the presentation, the design of the slides, and the speaker's gesture, posture, and speech quality. Many public speech training platforms are available in which verbal and non-verbal aspects of presentation are measured and feedback is provided. Batrinca et al. developed a public speech training platform with a virtual audience [5]. They found a correlation between automatically measured non-verbal descriptors of speech and expert assessments. Chen et al. [17] present a system for automatic assessment of public speaking skills using motion tracking and speech processing. Lui et al. [4] developed a mobile application which provides feedback on body motion, voice intensity and timing. AwareMe [18] measures voice pitch, filler words, and speaking rate during presentation practice and provides visual and haptic feedback through a wristband device. Trinh et al. [3] developed an integrated environment that supports structured presentation rehearsal.

Several speech assistance studies aim at providing support during presentation delivery. Saket et al. [19] designed a mobile application for timing support during the presentation. Tam et al. [20] developed a wireless wrist-worn system which provides haptic feedback for time-management. Rhema's system [21] provides visual feedback on speaking rate and volume using Google Glass. DynaimcDuo [6] is a PowerPoint plugin, which provides a virtual agent as a co-presenter. The agent can deliver pre-assigned parts of the presentation at the user's request. A user study showed that using this system resulted in significant decrease in users' anxiety, increase in their confidence, and higher audience ratings, compared to solo presentations.

2.1 Presentation Tracking

There have been limited studies on presentation tracking. Rogina et al. [9] developed a lecture tracker which can be used to switch slides and display the documents related to speech. They used dynamic time warping to match slides with the ASR output hypothesis. They achieved about 30% improvement in tracking error rate compared to a baseline method that assigned each slide a time slot of the same size. Okada et al. [10] computed the minimum distance between ASR hypothesis and speech script in order to track the current state of speech in real-time. The tracking system was designed for supporting master of ceremony (MC) performances and assumed that the speech performance would be very close to the planned speaking notes. I use a similar approach as my baseline method.

Some of the studies related to audio lecture indexing and retrieval use techniques which can be used for tracking. Lu et al. [11] used entropy-based word filtering, reliability-propagated word-based matching, and structured support vector machines to align utterance clusters with slide subsections. To my knowledge, this is the only work to date on alignment within slides; however, it assumes in-order presentation, and is an off-line system. In [22] Yamamoto et al. segmented lecture transcriptions into topics by associating them with the textbook used in the lecture. To do so, *tf.idf* vectors for text topics and speech transcripts were calculated, and then the cosine similarity between vectors was used for association.

The above studies show that the two main challenges in presentation tracking are speech recognition and text retrieval. In the following subsections, I will review the studies related to these two topics.

2.2 Speech Recognition

Speech recognition has improved significantly from single-speaker digit recognition systems in 1952 [23] to speaker-independent continuous speech recognition systems based on deep neural networks [24]. Currently, several open source ASR engines such as Pocketsphinx [25], Kaldi [26], and HTK [27] are

available, but accurate speech recognition requires high processing power which cloud based services such as IBM Watson [28], Google cloud platform [29] provide. Speech recognition has been used in different applications related to audio lectures including transcription, indexing, and retrieval [30, 31, 32]. Almost all of the speech recognition systems use acoustic and language models and have a vocabulary which contains the words that can be recognized by them [33]. Acoustic models provide a link between audio signals and the linguistic units like phonemes, and language model assigns probabilities to sequence of words, which is used for distinguishing between acoustically similar word sequences. The accuracy of the speech recognition is highly dependent on these models.

Previous studies show that the accuracy of the ASR can be improved for lecture transcription by retrieving text related to the presentation from the web or other supplementary material, and adapting the vocabulary and language model used for speech recognition. Park et al. [34] used a combination of spontaneous speech resources and textbooks as the language model. Munteanu et al. [35] used the contents of all the slides in the lecture to compile a corpus which eliminated the need for two different general and topic specific language models. Maergner et al. [36] used feature-based ranking for vocabulary selection. They generated a vocabulary using a collection of documents that were similar to lecture slides, and then ranked the resulting vocabulary based on a combination of word features.

For presentation tracking, I can't rely on high processing power since it should be possible to run the system on users' machines with minimal resources; therefore, I plan to use cloud based speech recognition systems which provide high accuracy with low cost. This results in limited control over the acoustic and language models used for ASR, and I can't use the approaches mentioned above to improve the accuracy of the ASR. Instead, I plan to reduce the vulnerability of systems to ASR errors by using confusion networks [13] instead of the best hypothesis (1-best). Fuji et al. [14] proposed using confusion networks to improve the robustness against recognition errors. They achieved 8.9% improvement in word error rate (WER) compared to using 1-best results. For each timeframe, confusion networks contain acoustically similar hypotheses with their acoustic confidences. This rich information has been utilized in many speech-related applications such as speech translation [37], semantic parsing [38], and spoken language understanding [39]. Hori et al. [40] used confusion networks for spoken utterance retrieval from MIT lecture corpus [41]. They performed keyword matching on out of vocabulary (OOV) words by combining phone and word confusion networks.

2.3 Semantic Text Retrieval

In order to measure the coverage of slide notes for presentation tracking, I need to associate the ASR output with related segments of notes. Text retrieval studies focus on matching queries against a set of text documents and can be useful for this task. Term weighting is important in text retrieval [42]. Salton et al. [43] compared different automatic term weighting methods for text retrieval. In their work, the query and document sets were represented by vectors containing all possible terms and their assigned weights. The similarity of query-documents was measured by vector similarity functions such as the cosine vector similarity formula. Different term frequencies, collection frequencies, and normalization factors were used for assigning weights to words. A normalization factor was used to remove the advantage of long documents which have higher term frequencies and more words. Singhal et al. [44] showed common normalization factors favor short documents in retrieval, and proposed a new normalization technique for retrieving documents based on their likelihood of relevance rather than their length. Term weighting is also the subject of studies on keyword extraction and text summarization.

There are several studies on summarizing and extracting keywords from lecture speech. Fuji et al. [45] automatically summarize lecture speech by extracting cue phrases using Conditional Random Fields (CRF). Kawahara et al. [46] extract characteristic keywords of the lecture using *tf-idf* and then use the extracted keywords as one of the measures for indexing key sentences in lecture archives. Yang et al. [47] only consider nouns and numbers as keyword candidates and then take the top N words ranked by the word frequency. Selected keywords were used for content-based lecture video search. Another useful approach is semantic retrieval of spoken keywords since speakers often utter words that are semantically related to

the text keywords without speaking any of the exact keywords [48]. In [49], the authors argued that keywords should be semantically relevant to the document theme and also provide a good coverage of the concepts. They clustered terms based on semantic relatedness and extracted key phrases from exemplar terms in these clusters. I used a similar approach to score keyword candidates.

Metrics for measuring the semantic similarity of words can be put in two categories [50]: corpus based measures, which use the information gathered from large corpora, e.g., word co-occurrence [51], and knowledge based measures, which use information from semantic networks such as WordNet [52]. Islam et al. [53] combined corpus based co-occurrence metric with string similarity metrics to measure the semantic similarity. Mihalcea et al. [50] define the semantic similarity between two text segments by combining the semantic similarities of each text segment with the other one. They identified and measured the similarity of the most similar word in each segment to each word in the other segment. Then, they used word specificity to assign weights to these similarity values and normalized them based on the length of sentence. Mikolov et al. [54] used continuous vector representations of words, computed from a very large dataset using neural networks, to measure word semantic similarity. The vectors representing words that are semantically close in the dataset are located close to each other in the vector space. Pennington et al. [55] proposed Global Vectors for word representation (GloVe) and claimed that their methods outperform other word embedding methods in text similarity tasks. GloVe vectors were trained using non-zero elements of a word-word co-occurrence matrix gathered from a large corpus. I propose using this method for measuring the semantic similarity in my proposed framework.

3. The Presentation Tracking Framework

In this section, I present a framework which uses confusion networks, semantic matching and keyword weighting to track the coverage of slide note contents during an oral presentation. The proposed framework consists of three main units: Slide Notes Processing, Speech Recognition, and Segment Tagging. At first, slide notes are segmented and keywords are extracted and weighted. At runtime, ASR system detects the slide keywords in speech and note segments are scored based on the detected keywords and their weights. Finally, segments that have scores higher than a threshold are tagged as covered. I plan to extend the framework to provide slide identification based on tagged segments. Figure 1 shows the overall architecture of the framework.

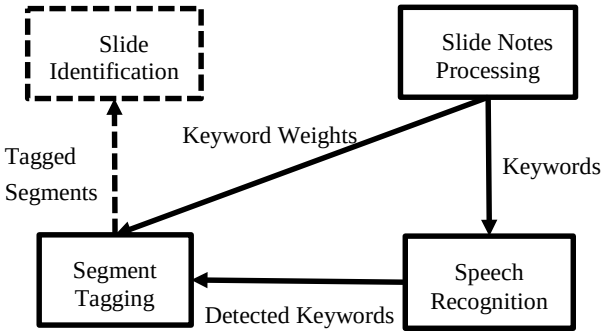

Figure 1: *Overall framework architecture. Dashed lines indicate future extensions.*

3.1 Slide Notes Processing

3.1.1 Note Segmentation

The slide notes should be segmented into smaller sections to make tracking more refined. However, if the sections are too small, such as at word level, overfitting might happen and even small deviations from the notes can result in false negative results. Some suggested methods for segmentations are sentence

segmentation, segmentation based on equal number of keywords in each segment, and semantic topic segmentation.

3.1.2 Keyword Extraction

In order to perform text matching, the framework extracts important words in each segment. Any of the keyword extraction methods mentioned in the previous section can be used for this purpose. Some of the common steps in keyword extraction are removing stop words and word lemmatization. The framework also extracts the synonyms for each keyword to remove the reliance on exact words. Synonyms can be extracted from on-line or off-line sources such as WordNet [52].

3.1.3 Keyword Weight Assignment

The framework assigns weights to keywords using two methods: *tf.idf* and semantic similarity.

Tf.idf weights: *As mentioned before tf.idf* is used for scoring the words based on their importance in a corpus of documents. I treat each notes segment as a document:

$$
\begin{aligned}
&tf.idf = tf(w_i, s_j) idf(w_i) \\
&tf(w_i, s_j) = frequncy\ of\ w_i\ in\ s_j \\
&idf(w_i) = \log\left(\frac{N}{n_t}\right) \\
&= \log\left(\frac{total\ number\ of\ sentences}{number\ of\ sentences\ containing\ w_i}\right)
\end{aligned}
\qquad (1)
$$

The reasoning behind this weighting is that if a word is used in multiple segments, it has low specificity for each of those segments. Therefore, compared to a unique word, it is less useful for detecting a segment. The issue with this method is that some unique words are not essential for the segment concept and thus can be omitted during the presentation. *tf.idf* gives high weights to such words, and if the word is ignored the weight for the segment might stay lower than the threshold which leads to false negative results.

Semantic similarity weights: To fix the above issue, I also consider the semantic similarity of keywords to the segments. Inspired by [49], I can argue that a good keyword should be more semantically relevant to the segment containing it compared to other segments. I model this concept using the similarity ratio score *sr*:

$$
sr(w_i) = \frac{local\ similairty(w_i)}{global\ similarity(w_i)} \qquad (2)
$$

Local similarity is the similarity of a segment keyword to other keywords in that segment. Global similarity is the similarity of a segment keyword to the keywords in other segments.

To measure the semantic similarity between words, I propose using GloVe vectors [55]. Word vectors representing more semantically similar words have smaller Euclidean distance and bigger cosine similarity. I will use both the Euclidean distance and cosine similarity of word vectors for calculating their semantic similarity.

To calculate the cosine similarity between a word and a word set containing it, I use the average cosine similarity between the word and all of the other words in that word set:

$$
\begin{aligned}
&\forall w_i \in W \qquad s_c(w_i, W) = \frac{\sum_{w_{j \neq i} \in W} sim(w_i, w_j)}{|W| - 1} \\
&sim(w_i, w_j) = \frac{\sum_{k=0}^{n} v(w_i)_k . v(w_j)_k}{\sqrt{\sum_{k=0}^{n} v(w_i)_k{}^2} \sqrt{\sum_{k=0}^{n} v(w_j)_k{}^2}}
\end{aligned}
\qquad (3)
$$

To calculate the similarity using the Euclidean distance I use a form of Closeness Centrality measure, which has been used in graph based key phrase extraction [56]:

$$\forall w_i \in W \quad s_d(w_i, W) = \frac{|W| - 1}{\sum_{w_{j \neq i} \in W} dist(w_i, w_j)}$$

$$dist(w_i, w_j) = \sqrt{\sum_{k=0}^{n} (v(w_i)_k - v(w_j)_k)^2} \qquad (4)$$

The value of *n* in equations 3 and 4 is equal to the number word vector dimensions. Finally, the similarity ratio in equation 3 is calculated using the cosine similarity or Euclidean distance:

$$sr(w_i) = \frac{s_c(w_i, W_L)}{s_c(w_i, W_G)} \; or \; \frac{s_d(w_i, W_L)}{s_d(w_i, W_G)} \qquad (5)$$

W_L is the set of words in the segment containing w_i and W_G is the set of words in other segments. Similarity ratios are used as coefficients of *tf.idf* weights and these combined weights are normalized by dividing the weight of each keyword by the sum of weights of all of the keywords in the segment:

$$\forall w_i \in W_{s_j} \;\; sw(w_i) = \frac{sr(w_i) tf.idf(w_i)}{\sum_{w_k \in W_{s_j}} sr(w_k) tf.idf(w_k)} \qquad (6)$$

Sample weighting scenario:

1. The **tiger** is the **largest cat species**.
2. An **adult male wild tiger** can **reach** a **total body length** of up to **11.5 feet**, and **weigh** up to **850 pounds**.
3. The **most common color** of **tigers** is **orange**, with **black stripes**.
4. But each **tiger** has a **unique stripe pattern**, much like our **fingerprints**.
5. We have also **seen** some **color variations**, with **white**, **black**, **golden tabby**, and **blue tigers**.
6. The **current population** of **wild tigers** is **estimated** to be about **3200 individuals**.
7. There are **10 recognized tiger subspecies**, but **four** of them are **considered extinct**.

Figure 2*: Sample slide notes with keyword candidates in bold.*

	tf.idf	sr_c	sr_d	sw_c	sw_d
orange	0.845	1.113	1.3	0.293	0.284
common	0.845	0.668	1.106	0.176	0.242
color	0.477	1.147	1.295	0.170	0.160
black	0.477	1.086	1.281	0.161	0.158
stripes	0.477	1.347	1.267	0.200	0.156
tiger	0	0.887	1.15	0	0

Table 1: *Keywords from segment 3 of Figure 2. sr_c and sr_d are similarity ratios using distance and cosine, sw_d and sw_c are normalized word weights using sr_d and sr_c*

To clarify the scoring process, I present a sample keyword scoring scenario. Figure 2 shows the notes for a sample slide with 7 segments. The keywords are in bold. Table 1 shows the weights for keywords of

segments in figure 2. The table is ordered by normalized similarity weights. We can see that *tf.idf* discards the word "tiger" since it is used in all segments. Similarity weights give higher weights to "orange", "black" and "color" since they are semantically close to each other and represent the main segment concept. The word "common" is weighted highly by *tf.idf* since it is only used in this segment but it has the lowest weight in similarity ratios. Normalizing the weights results in lowering the final weight for "common". This effect is more evident in sw_c, which uses the cosine similarity weight.

3.2 Speech Recognition

The framework uses confusion networks to spot the slide note keywords in speech. Some ASR systems provide confusion networks as an output option but if not available the ASR lattice output can be decoded to generate the confusion networks graph [13]. The confusion network contains alternative words in each time frame ordered by acoustic confidence score. The framework will iterate through this ordered list and compare each word and its synonyms with the keyword candidates and their synonyms. If there is a match, other alternatives in that time slot are discarded and the matched keyword candidate is tagged as spotted in all segments containing that candidate. Each candidate can only be spotted once in a slide. Figure 3 shows a sample keyword spotting scenario.

ASR output sorted based on acoustic confidence level

Confidence	*Alternative*	*Synonyms*
0.62	variation	fluctuation
0.31	alteration	change, **modification**
0.06	operation	

"modification" is matched

Sample keyword candidates

Keyword	*Synonyms*
photo	photograph
color	colour
adjustment	**modification**
crucial	important

"adjustment" is spotted

Figure 3: *A sample keyword spotting scenario.*

3.3 Segment Tagging

Each time a keyword is spotted the score for the segments containing it will increase by the amount equal to the weight for that keyword. Therefore:

$$ss(s_i) = \sum_{w_j \in W_{s_i}} c(w_j) sw(w_j) \qquad (7)$$

Where $ss(s_i)$ is the coverage score for the segment s_i and $sw(w_j)$ is the weight for keyword w_j in W_{s_i} which is the set of keyword in s_i. $c(w_j)$ is equal to 1 when w_j is spotted, otherwise it is 0. When the score for a segment is higher than a threshold, that segment will be tagged as covered.

3.4 Slide Identification

The framework will also identify the slide that the speaker is presenting, given the slide notes and a representation of the logical relationship among them (using sequential partial ordering and decomposition relations). This enables the framework to support more extemporaneous and dynamic presentations. Previous studies have examined methods for off-line alignment of lecture speech transcripts with slides [64, 65]. For on-line tracking, I plan to utilize similar semantic keyword matching methods used in sub-slide level tracking and also investigate the efficacy of using additional knowledge sources to improve tracking accuracy.

Inspired by previous studies [66, 67], I can describe the interaction between the tracking framework and the user as a collaborative process, with the user dynamically initiating presentation about different topics and the framework displaying the proper slides. Plan recognition models [68] can be used for human-agent collaboration by exploiting hierarchical task plans [69]. In a similar approach, I plan to use the information about the composition of the presentation and the precedence relationship of slides to determine the set of

slides with the highest probability of being presented at each moment. This hierarchical structural information can be provided by the user in addition to slide notes and will include the sequence of slides in each section and subsection of the presentation.

4. Pilot Implementation

I have developed and evaluated a preliminary implementation of the framework.

4.1 Implementation Details

4.1.1 Keyword Extraction

The system uses Stanford CoreNLP tools [57] for segmenting the slide notes into sentences, and performing the part of speech tagging. It removes stop words, punctuation marks and symbols and converts numbers into their word representations. The remaining words are lemmatized and extracted as keywords. WordNet [52] is used to extract the synonyms for each keyword. To do so, the most common synset for each word is retrieved and the words in that synset are extracted as that word's synonyms. If a word is a synonym for multiple words, the word in the synset with highest tagged frequency in WordNet is chosen. WordNet stemming is used in addition to CoreNLP lemmatization for comparative and superlative adjectives. The keyword candidates in their base form and their synonyms are stored in a table.

4.1.2 Keyword Weighting

Keywords are weighted based on semantic similarity and *tf.idf* measures. For semantic similarity weighing the system uses a pre-trained GloVe vector representation with 1.9 million uncased words and vectors with 300 elements. It was trained using 42 billion tokens of web data from Common Crawl.

4.1.3 Speech Recognition

Automatic speech recognition is performed using IBM's Watson cloud-based service [28]. This service provides both n-best transcripts and confusion networks. The tracking system discards confusion network alternative hypotheses with the level of confidence lower than 0.01. The threshold is chosen based on trial and error.

4.2 Evaluation Experiments

I evaluated this system using a corpus of 30 videotaped presentations delivered by 15 speakers (6 female, 9 male, 13 non-native English speakers) on the topics of lions and tigers. Each presentation contained 5 slides, with an average length of 5 minutes. Each slide had detailed speaking notes containing 6-10 sentences with an average of 8 sentences. Recordings were split for each slide, resulting in 150 slide presentation recordings. In order to simulate real-time tracking conditions, each slide presentation recording was segmented into 3 sections in a semi-random manner. This was done by detecting the 2 longest pauses in speech and splitting the recording around them. I also made sure that each segment is at least 6 seconds long. Using this segmentation method, each speech segment might cover a random number of sentences from zero to the total number of sentences in the slide. A few recordings were too short to be split and were discarded. This process resulted in a total of 426 audio files.

Each audio file was manually annotated for content coverage by a human annotator. The annotator was instructed to subjectively tag a sentence as covered if she found that the main points of the sentence were covered in sufficient detail. 100 recordings were randomly chosen and annotated by another annotator to check the inter-rater agreement. The Cohen's Kappa coefficient was 84%, which indicates a high degree of agreement. In order to evaluate the results, the files were also automatically annotated using different tracking methods and thresholds. Manual and automatic annotations were compared and precision, recall, and f-score measures were calculated for each method.

I used a tracking method similar to [10] as the baseline. In this method the 1-best ASR results were matched against the slide notes. Note sentences were scored based on the ratio of the spotted keywords to

total number of the keywords in sentence. Table 2 lists the evaluated tracking methods and their reference names in the Results subsection.

Name	Description
baseline	Using 1-best ASR output
synons	*baseline* + synonyms
words	confusion network ASR output + synonyms
tfidf	*words* method + tf.idf score
cosine	*tfidf* method + cosine similarity weighting
distance	*tfidf* method + Euclidean distance weighting

Table 2: *Evaluated tracking methods' reference names and their descriptions*

4.3 Results

Figure 4 shows the precision-recall curves for 3 tracking methods with thresholds changing from 0 to 1. It also includes the curves for the highest F-score values for *distance* and *baseline* methods. Precision, recall and F-score values have increased compared to the baseline method. Increasing the threshold generally results in lower recall but higher precision values. In this case, threshold values between 0.2 and 0.3 led to the best F-scores for all methods. The unsupervised nature of the system and different application requirements discourage me from determining an optimal threshold for all applications.

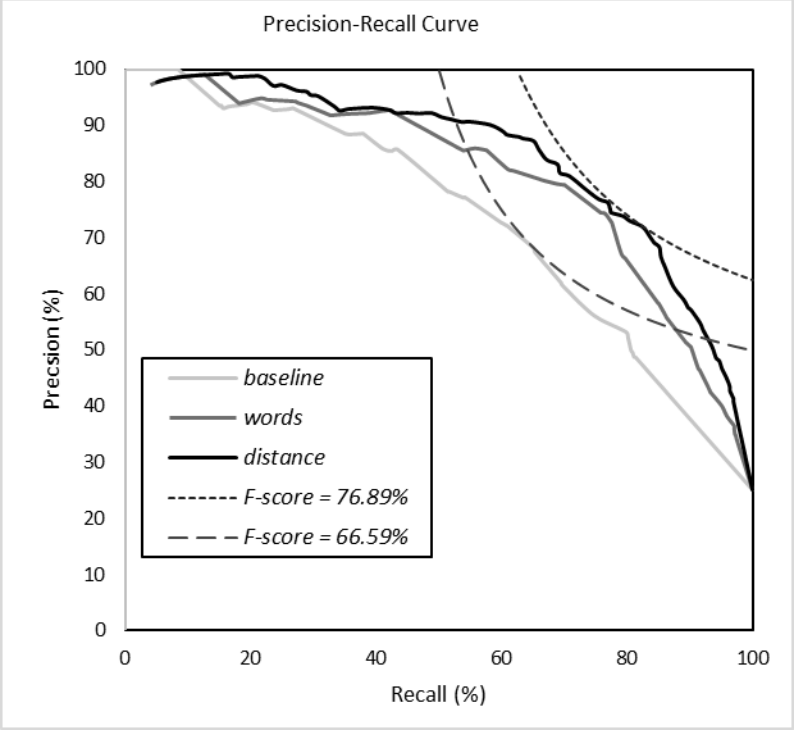

Figure 4: *Precision-Recall curves for different tracking methods*

Table 3 shows the values for precision, recall and F-score for each experiment using threshold values optimized for F-score. An approach that randomly tags sentences as covered is also included in the table for comparison. We can see the positive effect of using synonyms and confusion networks in improvements from *baseline* to *synons* and from *synons* to *words* method.

Method Name	Precision (%)	Recall (%)	F-score (%)
random	25.53	53.44	34.55
baseline	68.13	65.11	66.59
synons	70.32	71.06	70.69
words	74.30	76.55	75.40
tfidf	76.24	76.78	76.51
cosine	**79.70**	73.75	76.61
distance	72.00	**82.50**	**76.89**

Table 3: *Evaluation measures using thresholds optimized for best F-score*

Using keyword weighting improves the F-score compared to the *words* method and the similarity weighting methods have the best F-scores. Using the Euclidean distance, similarity weighting results in the best recall value and the *cosine* method has the best precision. Depending on the application requirement, we can choose between these two similarity weighting methods.

The results demonstrate that using confusion networks, semantic matching and keyword weighting improve tracking accuracy.

5. Proposed Work and Schedule

I plan to investigate different solutions for text segmentation and keyword weighting to further improve the accuracy of the tracking framework. Also I will extend the framework to include slide tracking. I plan to evaluate the effectiveness of the presentation tracking framework in presentation rehearsal and delivery applications.

5.1 Improving the Accuracy and Extending the Framework

5.1.1 Notes Segmentation

As mentioned in section 3.1.1, the method used for segmentation of the notes affects the accuracy and usability of the tracking system. In the evaluation study, the notes were segmented into sentences which resulted in segments with variable lengths and different number of keywords. This results in unequal probability of detection for different segments. Segmenting the notes into coherent topics can result in more similar numbers of keywords and fix this issue. Automatic topic segmentation has been the focus of many studies in natural language processing [58, 59, 60]. I plan to review these studies and examine the effect of using different automatic topic segmentation methods on the performance of my tracking framework.

5.1.2 Keyword Weighting

The evaluation study showed that keyword weighting can improve the accuracy of tracking. I used *tf.idf* and semantic similarity methods to assign weights to keywords, but several other term weighting strategies have been explored in information retrieval [61], sentiment analysis [62], and text classification [63] studies, which might result in further improvements in tracking accuracy. I will survey other keyword weighting methods and evaluate the accuracy of tracking using these methods.

5.1.3 Slide Level Tracking

As discussed in Section 3.4, I will extend my framework to also identify the slide that the speaker is presenting. I will evaluate the efficacy of using the relations of slides in addition to keyword matching results for providing slide level tracking.

5.2 Evaluating the Effectiveness in a Rehearsal Application

Prior presentation rehearsal systems can provide instant or delayed feedback on presentation quality [4, 5, 18], which have been shown to be effective for improving public speech performance [70]. However, these

systems do not provide feedback on sub-slide segments of the presentation content. I propose using my presentation tracking framework to align the tracking results with the speech quality measurements through time to connect presentation notes segments with their related speech quality measurements. This will allow the rehearsal system to provide feedback on the presentation quality of each topic to the user. I plan to investigate the effectiveness providing speech quality feedback on sub-slide segments, compared to feedback on entire slide or presentations.

5.2.1 Presentation Rehearsal System

To automatically assess the speech quality, I plan to measure common speech quality features used in previous speech quality assessment systems, including pitch, speaking rate, and filler word occurrences [17, 71, 72]. Pitch or fundamental frequency of speech is one of the prosodic features of speech which is related to the rate of vibrations of the vocal cord [73]. Higher pitch variations are correlated with better speech quality [5]. Speaking rate is important for speech comprehension [74]. Previous studies have tried to set proper speaking rate ranges for different tasks [75]. Filled pauses or filler sounds (e.g., *uh* /ʌ/, *er* /ɜ:/, and *um* /ʌm/) are one of the most common indicators of disfluency, anxiety, and hesitation in speech [76].

I have developed tools for measuring these metrics in Praat [77]. I extracted the pitch and intensity contours and calculated the mean, range, and standard deviation values. I used the method in [78] to estimate the speaking rate. To do so, I extracted the voiced part of the speech by identifying sections in the signal with the intensity values of at most 25dB lower than 0.99 quantile maximum intensity and pitch values higher than 100Hz. These values are default Pratt settings for pause detection. Peaks in the intensity envelope of the voiced parts of signal were identified and the ones that were at least 2 dB higher than their succeeding peaks were extracted as syllable nuclei. Figure 5 shows a sample signal with its pitch and intensity contours and the detected syllable nuclei. To calculate the speaking rate, I divided the number of syllables by the total speaking time including the pauses.

To identify the filler sounds in speech, I extracted the first, second and third formant frequencies and used the method described in [79]. A time window of length 100 millisecond was moved along the signal in steps of 20 milliseconds. The standard deviations of the formant frequencies were calculated in this window. Regions of voiced speech signal that had formant frequency deviations less than 100 Hz were tagged as filled pauses.

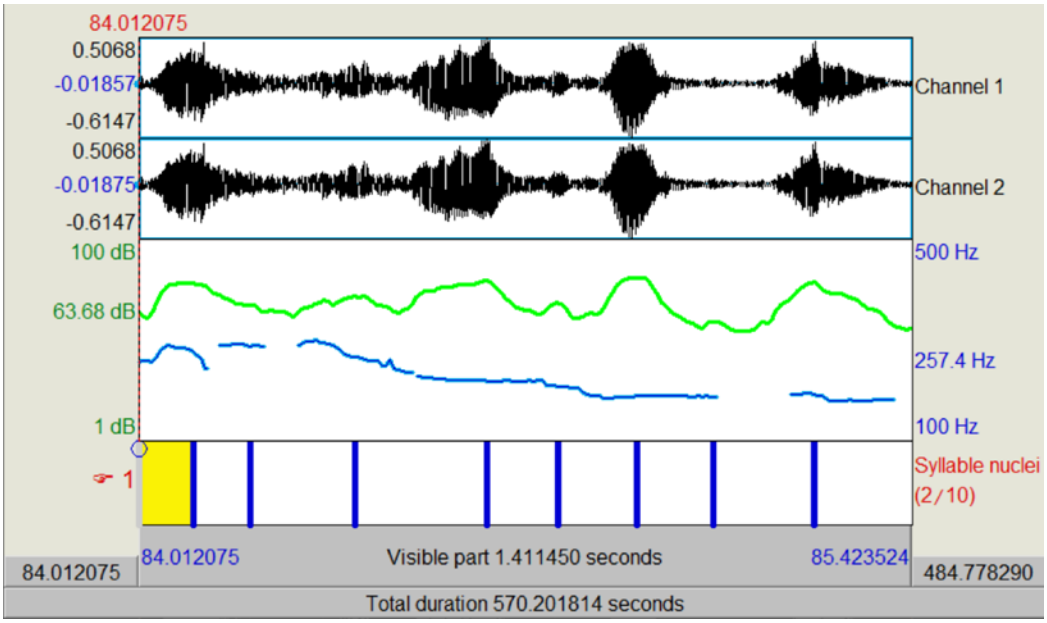

Figure 5. *Syllable nuclei extraction in Praat: Top most tier shows the signal waveform, Middle tier shows the intensity (green) and pitch (blue) contours, Bottom pane show the extracted syllable nuclei*

5.2.2 User Study

I plan to conduct a within-subjects user study to examine the effectiveness of my tracking framework in rehearsal systems. The study will include two conditions: presentation rehearsal using feedback on entire slides or presentations, and rehearsal using the feedback on sub-slide segments. In each condition the participants will be provided with a set of slides and asked to practice their presentations using the proposed or the control rehearsal systems and then deliver their presentation. The participants will present different set of slides for each session and the order of the presentation slide sets and conditions will be randomized and counter balanced. The slides will include detailed notes which will be used by the tracking framework.

The presentations will be recorded and the participants will be asked to rate the rehearsal system using a 6-question, 7-points scale measure. I will also recruit another set of participants as judges to watch the recorded presentations and rate the relative quality of speech in both sessions for each participant.

I hypothesize that:

1- The speakers will rate the sub-slide rehearsal system higher than the control system.
2- The presentations delivered after the sub-slide rehearsal system will get higher quality ratings from the judges.

5.3 Evaluating the Effectiveness in a Presentation Delivery Application

5.3.1 The Dynamic-Duo Co-Presenter System

I plan to integrate my framework into Dynamic-Duo co-presenter system [6]. Dynamic-Duo provides a virtual agent as a co-presenter in an integrated environment. The virtual agent, is a human-like animated character with synthesized voice and non-verbal behaviors. The system provides note-authoring tools for the agent and human presenter. The slide notes are segmented into multiple subsections and these subsections can be assigned to the human presenter or the agent. Figure 7 shows the note authoring environment. The user interacts with the system using a remote control ("clicker") to change the slides or ask Angela to read her assigned subsections.

Figure 6. *Dynamic-Duo co-presentation system*

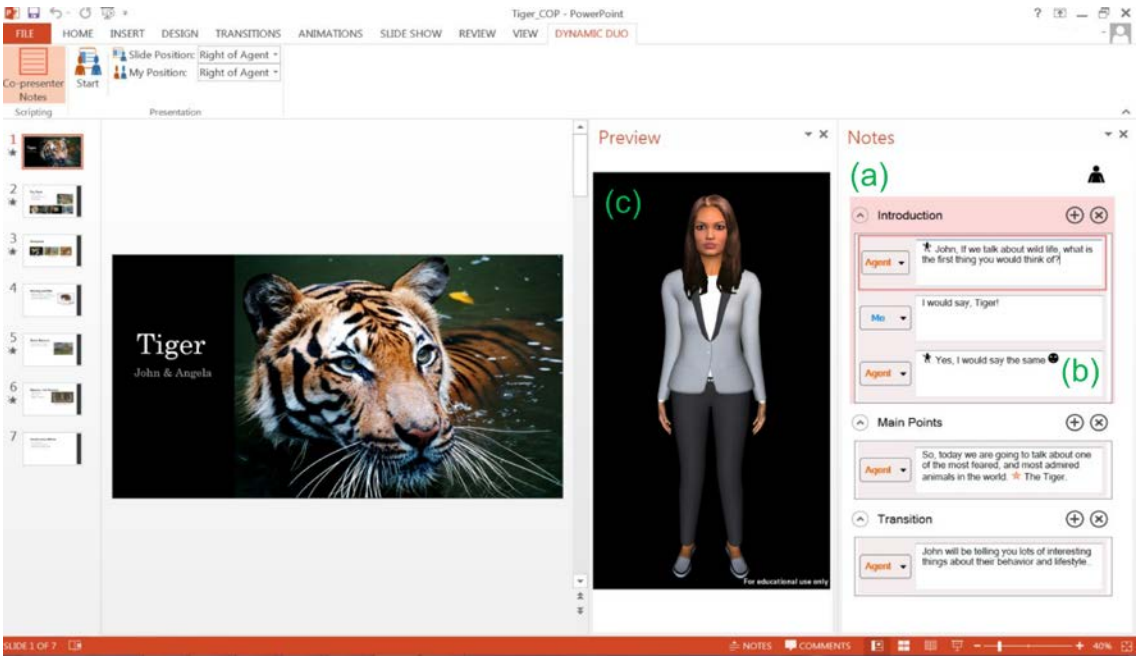

Figure 7. *Dynamic-Duo note authoring environment: a) Notes for agent and human presenter. b) A note subsection assigned to the agent. c) Preview of the agent.*

I propose a new interaction system for this co-presenter system using the tracking framework. In this new system, anytime during the presentation, the speaker can use the clicker to ask the agent to continue the presentation, in which case, the tracking system will detect the covered sections of the slide and the agent will present the uncovered sections. This can result in less pre-planned presentations since the user will be able to present the sections assigned to the agent or ask the agent to cover the sections that were assigned to the human speaker. The system will also be helpful in cases that the users forget their assigned sections of the presentation. I propose that the speakers will have less anxiety during the presentation using this system and the overall presentation will cover more content.

5.3.2 User Study

I will conduct a within subject user study to examine the effectiveness of the proposed interaction system. The study will include two conditions, the co-presenter with tracking capabilities and co-presenter without tracking. In each condition the participants will be provided with a set of slides and asked to practice and then deliver their presentations using the control or proposed co-presenter systems. The participants will use different set of slides for each session and the order of the presentation slide sets and conditions will be randomized and counter-balanced. The slides will include detailed notes which will be used by the tracking framework.

The presentations will be recorded and the participants will be asked to fill-out questionnaires on their anxiety levels before the presentations and rate the co-presenter system after the presentation using a 6-question, 7-point scale measure. The recorded presentations will be analyzed and the content coverage will be measured. I will also recruit another set of participants as judges to watch the recorded presentations and rate the relative quality of speech in both sessions for each participant.

I hypothesize that:

1- The speakers will report less anxiety before their presentations with the proposed co-presenter system.
2- The speakers will rate the proposed co-presenter system higher than the control system.
3- The content coverage in the proposed system will be higher.
4- The presentations delivered using the proposed system will get higher quality ratings from the judges.

5.4 Tentative Schedule

Improving the tracking accuracy	August 2016
Rehearsal user study	September 2016
Slide tracking	October 2016
Co-presenter study	November - December 2016
Writing the dissertation	January - March 2017
Thesis defense	April 2017

References

1. Microsoft PowerPoint. http://office.microsoft.com/enus/powerpoint.
2. Apple Keynote. http://www.apple.com/iwork/keynote.
3. Trinh, Ha, Koji Yatani, and Darren Edge. "PitchPerfect: integrated rehearsal environment for structured presentation preparation." *Proceedings of the 32nd annual ACM conference on Human factors in computing systems*. ACM, 2014.
4. Lui, Andrew Kwok-Fai, Sin-Chun Ng, and Wing-Wah Wong. "A Novel Mobile Application for Training Oral Presentation Delivery Skills." *Technology in Education. Technology-Mediated Proactive Learning*. Springer Berlin Heidelberg, 2015. 79-89.
5. Batrinca, Ligia, et al. "Cicero-towards a multimodal virtual audience platform for public speaking training." *Intelligent Virtual Agents*. Springer Berlin Heidelberg, 2013.
6. Trinh, H., Ring, L., & Bickmore, T. (2015, April). DynamicDuo: Co-presenting with Virtual Agents. In Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (pp. 1739-1748). ACM.
7. Nijholt, Anton, Herwin van Welbergen, and Job Zwiers. "Introducing an Embodied Virtual Presenter Agent in a Virtual Meeting Room." *Artificial Intelligence and Applications*. 2005.
8. Goodman, A. (2006). *Why bad presentations happen to good causes, and how to ensure they won't happen to yours*. Cause Communications.
9. Rogina, I., & Schaaf, T. (2002). Lecture and presentation tracking in an intelligent meeting room. In Multimodal Interfaces, 2002. Proceedings. Fourth IEEE International Conference on (pp. 47-52). IEEE.
10. Okada, T., Yamamoto, T., Terada, T., & Tsukamoto, M. (2011, March). Wearable MC system a system for supporting MC performances using wearable computing technologies. In Proceedings of the 2nd Augmented Human International Conference (p. 25). ACM.
11. Lu, H., Shen, S. S., Shiang, S. R., Lee, H. Y., & Lee, L. S. (2014). Alignment of Spoken Utterances with Slide Content for Easier Learning with Recorded Lectures using Structured Support Vector Machine (SVM). In Fifteenth Annual Conference of the International Speech Communication Association.
12. Sainath, T. N., Vinyals, O., Senior, A., & Sak, H. (2015, April). Convolutional, long short-term memory, fully connected deep neural networks. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on* (pp. 4580-4584). IEEE.
13. Mangu, Lidia, Eric Brill, and Andreas Stolcke. "Finding consensus in speech recognition: word error minimization and other applications of confusion networks." Computer Speech & Language 14.4 (2000): 373-400.
14. Fujii, Y., Yamamoto, K., & Nakagawa, S. (2010, September). Improving the readability of class lecture ASR results using a confusion network. In *INTERSPEECH* (pp. 3078-3081).
15. Carpineto, C., & Romano, G. (2012). A survey of automatic query expansion in information retrieval. *ACM Computing Surveys (CSUR)*, *44*(1), 1.
16. Ramos, J. (2003, December). Using tf-idf to determine word relevance in document queries. In *Proceedings of the first instructional conference on machine learning*.

17. Chen, Lei, et al. "Towards automated assessment of public speaking skills using multimodal cues." *Proceedings of the 16th International Conference on Multimodal Interaction*. ACM, 2014.
18. Bubel, M., Jiang, R., Lee, C. H., Shi, W., & Tse, A. (2016, May). AwareMe: Addressing Fear of Public Speech through Awareness. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems* (pp. 68-73). ACM.
19. Saket, B., Yang, S., Tan, H., Yatani, K., & Edge, D. (2014, September). Talkzones: Section-based time support for presentations. In *Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services* (pp. 263-272). ACM.
20. Tam, D., MacLean, K. E., McGrenere, J., & Kuchenbecker, K. J. (2013, April). The design and field observation of a haptic notification system for timing awareness during oral presentations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 1689-1698). ACM.
21. Tanveer, M. Iftekhar, Emy Lin, and Mohammed Ehsan Hoque. "Rhema: A real-time in-situ intelligent interface to help people with public speaking."*Proceedings of the 20th International Conference on Intelligent User Interfaces*. ACM, 2015.
22. Yamamoto, N., Ogata, J., & Ariki, Y. (2003, September). Topic segmentation and retrieval system for lecture videos based on spontaneous speech recognition. In INTERSPEECH.
23. Juang, B. H., & Rabiner, L. R. (2005). Automatic speech recognition–a brief history of the technology development. *Georgia Institute of Technology. Atlanta Rutgers University and the University of California. Santa Barbara*, *1*, 67.
24. Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A. R., Jaitly, N., ... & Kingsbury, B. (2012). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, *29*(6), 82-97.
25. Huggins-Daines, David, et al. "Pocketsphinx: A free, real-time continuous speech recognition system for hand-held devices." *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*. Vol. 1. IEEE, 2006.
26. Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., ... & Silovsky, J. (2011). The Kaldi speech recognition toolkit. In *IEEE 2011 workshop on automatic speech recognition and understanding* (No. EPFL-CONF-192584). IEEE Signal Processing Society.
27. Woodland, P. C., Odell, J. J., Valtchev, V., & Young, S. J. (1994, April). Large vocabulary continuous speech recognition using HTK. In *Acoustics, Speech, and Signal Processing, 1994. ICASSP-94., 1994 IEEE International Conference on* (Vol. 2, pp. II-125). IEEE.
28. "Speech to Text | IBM Watson Developer Cloud", ibm.com, 2016. [Online]. Available: http://www.ibm.com/smarterplanet/us/en/ibmwatson/developercloud/speech-to-text.html. [Accessed: 30- Mar- 2016].
29. "Speech API - Speech Recognition | Google Cloud Platform", google.com, 2016. [Online]. Available: https://cloud.google.com/speech/. [Accessed: 15- June- 2016].
30. Glass, J. R., Hazen, T. J., Cyphers, D. S., Malioutov, I., Huynh, D., & Barzilay, R. (2007, August). Recent progress in the MIT spoken lecture processing project. In *Interspeech* (pp. 2553-2556).
31. Repp, S., & Meinel, C. (2006, March). Semantic indexing for recorded educational lecture videos. In *Pervasive Computing and Communications Workshops, 2006. PerCom Workshops 2006. Fourth Annual IEEE International Conference on* (pp. 5-pp). IEEE
32. Togashi, S., & Nakagawa, S. (2008, September). A browsing system for classroom lecture speech. In *INTERSPEECH* (pp. 2803-2806).Rogina, I., & Schaaf, T. (2002). Lecture and presentation tracking in an intelligent meeting room. In Multimodal Interfaces, 2002. Proceedings. Fourth IEEE International Conference on (pp. 47-52). IEEE.
33. Rabiner, L., & Juang, B. H. (1993). Fundamentals of speech recognition.
34. Park, A., Hazen, T. J., & Glass, J. R. (2005, March). Automatic Processing of Audio Lectures for Information Retrieval: Vocabulary Selection and Language Modeling. In ICASSP (1) (pp. 497-500).
35. Munteanu, C., Penn, G., & Baecker, R. (2007, August). Web-based language modelling for automatic lecture transcription. In INTERSPEECH (pp. 2353-2356).
36. Maergner, P., Waibel, A., & Lane, I. (2012, March). Unsupervised vocabulary selection for real-time speech recognition of lectures. In Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on (pp. 4417-4420). IEEE.
37. Bertoldi, N., Zens, R., & Federico, M. (2007, April). Speech translation by confusion network decoding. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07* (Vol. 4, pp. IV-1297). IEEE.

38. Tür, Gökhan, Anoop Deoras, and Dilek Hakkani-Tür. "Semantic parsing using word confusion networks with conditional random fields." *INTERSPEECH*. 2013.
39. Henderson, Mike, et al. "Discriminative spoken language understanding using word confusion networks." *Spoken Language Technology Workshop (SLT), 2012 IEEE*. IEEE, 2012.
40. Hori, T., Hetherington, I. L., Hazen, T. J., & Glass, J. R. (2007, April). Open-vocabulary spoken utterance retrieval using confusion networks. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07* (Vol. 4, pp. IV-73). IEEE.
41. Glass, J., Hazen, T. J., Hetherington, L., & Wang, C. (2004, May). Analysis and processing of lecture audio data: Preliminary investigations. In *Proceedings of the Workshop on Interdisciplinary Approaches to Speech Indexing and Retrieval at HLT-NAACL 2004* (pp. 9-12). Association for Computational Linguistics.
42. Buckley, C. (1993, March). The importance of proper weighting methods. In *Proceedings of the workshop on Human Language Technology* (pp. 349-352). Association for Computational Linguistics.
43. Salton, G., & Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information processing & management*, *24*(5), 513-523.
44. Singhal, A., Buckley, C., & Mitra, M. (1996, August). Pivoted document length normalization. In *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 21-29). ACM.
45. Fujii, Y., Kitaoka, N., Nakagawa, S., & Nakagawa, S. (2007, August). Automatic extraction of cue phrases for important sentences in lecture speech and automatic lecture speech summarization. In INTERSPEECH (pp. 2801-2804).
46. Kawahara, T., Shitaoka, K., Kitade, T., & Nanjo, H. (2003). Automatic indexing of key sentences for lecture archives. In Automatic Speech Recognition and Understanding, 2003. ASRU'03. 2003 IEEE Workshop on (pp. 141-144). IEEE.
47. Yang, H., & Meinel, C. (2014). Content based lecture video retrieval using speech and video text information. Learning Technologies, IEEE Transactions on, 7(2), 142-154.
48. Lee, L. S., Glass, J., Lee, H. Y., & Chan, C. A. (2015). Spoken Content Retrieval—Beyond Cascading Speech Recognition with Text Retrieval. Audio, Speech, and Language Processing, IEEE/ACM Transactions on, 23(9), 1389-1420.
49. Liu, Z., Li, P., Zheng, Y., & Sun, M. (2009, August). Clustering to find exemplar terms for keyphrase extraction. In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1 (pp. 257-266). Association for Computational Linguistics.
50. Mihalcea, R., Corley, C., & Strapparava, C. (2006, July). Corpus-based and knowledge-based measures of text semantic similarity. In *AAAI* (Vol. 6, pp. 775-780).
51. Turney, P. D. (2002, July). Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th annual meeting on association for computational linguistics* (pp. 417-424). Association for Computational Linguistics.
52. Miller, G. A. (1995). WordNet: a lexical database for English. Communications of the ACM, 38(11), 39-41.
53. Islam, A., & Inkpen, D. (2008). Semantic text similarity using corpus-based word similarity and string similarity. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, *2*(2), 10.
54. Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.
55. Pennington, J., Socher, R., & Manning, C. D. (2014, October). Glove: Global Vectors for Word Representation. In *EMNLP* (Vol. 14, pp. 1532-43).
56. Boudin, Florian. "A comparison of centrality measures for graph-based keyphrase extraction." *International Joint Conference on Natural Language Processing (IJCNLP)*. 2013.
57. Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., & McClosky, D. (2014, June). The Stanford CoreNLP Natural Language Processing Toolkit. In ACL (System Demonstrations) (pp. 55-60).
58. Eisenstein, J., & Barzilay, R. (2008, October). Bayesian unsupervised topic segmentation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (pp. 334-343). Association for Computational Linguistics.
59. Du, L., Buntine, W. L., & Johnson, M. (2013). Topic Segmentation with a Structured Topic Model. In *HLT-NAACL* (pp. 190-200).
60. Jameel, S., & Lam, W. (2013, July). An unsupervised topic segmentation model incorporating word order. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval* (pp. 203-212). ACM.

61. Blanco, R., & Lioma, C. (2012). Graph-based term weighting for information retrieval. *Information retrieval*, *15*(1), 54-92.
62. Deng, Z. H., Luo, K. H., & Yu, H. L. (2014). A study of supervised term weighting scheme for sentiment analysis. *Expert Systems with Applications*,*41*(7), 3506-3513.
63. Ko, Y. (2012, August). A study of term weighting schemes using class information for text classification. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval* (pp. 1029-1030). ACM.
64. Swaminathan, R., Thompson, M. E., Fong, S., Efrat, A., Amir, A., & Barnard, K. (2010, August). Improving and aligning speech with presentation slides. In*Pattern Recognition (ICPR), 2010 20th International Conference on* (pp. 3280-3283). IEEE.
65. Chen, Y., & Heng, W. J. (2003, May). Automatic synchronization of speech transcript and slides in presentation. In *Circuits and Systems, 2003. ISCAS'03. Proceedings of the 2003 International Symposium on* (Vol. 2, pp. II-568). IEEE.
66. Ferguson, G., & Allen, J. F. (1998, July). TRIPS: An integrated intelligent problem-solving assistant. In *AAAI/IAAI* (pp. 567-572).
67. Litman, D. J., & Allen, J. F. (1987). A plan recognition model for subdialogues in conversations. *Cognitive science*, *11*(2), 163-200.
68. Carberry, S. (2001). Techniques for plan recognition. *User Modeling and User-Adapted Interaction*, *11*(1-2), 31-48.
69. Lesh, N., Rich, C., & Sidner, C. L. (1999). Using plan recognition in human-computer collaboration. In UM99 User Modeling (pp. 23-32). Springer Vienna.
70. King, P. E., Young, M. J., & Behnke, R. R. (2000). Public speaking performance improvement as a function of information processing in immediate and delayed feedback interventions. *Communication Education*,*49*(4), 365-374.
71. Chen, Lei, et al. "Using multimodal cues to analyze mla'14 oral presentation quality corpus: Presentation delivery and slides quality." *Proceedings of the 2014 ACM workshop on Multimodal Learning Analytics Workshop and Grand Challenge*. ACM, 2014.
72. Brilman, Maarten, and Stefan Scherer. "A Multimodal Predictive Model of Successful Debaters or How I Learned to Sway Votes." *Proceedings of the 23rd Annual ACM Conference on Multimedia Conference*. ACM, 2015.
73. Hess, W. (2012). *Pitch determination of speech signals: algorithms and devices* (Vol. 3). Springer Science & Business Media.
74. Anderson-Hsieh, J., & Koehler, K. (1988). The effect of foreign accent and speaking rate on native speaker comprehension. *Language learning*, *38*(4), 561-613.
75. Venkatagiri, H. S. (1999). Clinical measurement of rate of reading and discourse in young adults. *Journal of Fluency Disorders*, *24*(3), 209-226.
76. Rose, R. L. (1998). *The communicative value of filled pauses in spontaneous speech* (Doctoral dissertation, University of Birmingham).
77. Boersma, Paul, and Vincent van Heuven. "Speak and unSpeak with PRAAT."*Glot International* 5.9-10 (2001): 341-347.
78. De Jong, Nivja H., and Ton Wempe. "Praat script to detect syllable nuclei and measure speech rate automatically." *Behavior research methods* 41.2 (2009): 385-390.
79. Audhkhasi, Kartik, et al. "Formant-based technique for automatic filled-pause detection in spontaneous spoken English." *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*. IEEE, 2009.