Using Term Clouds to Represent Segment-Level Semantic Content of Podcasts

Marguerite Fuller
Centre for Digital Video Processing
Dublin City University, Ireland
mfuller@computing.dcu.ie

Manos Tsagkias
ISLA, University of Amsterdam
e.tsagkias@uva.nl

Eamonn Newman
Centre for Digital Video Processing
Dublin City University, Ireland
enewman@computing.dcu.ie

Jana Besser
ISLA, University of Amsterdam
jbesser@science.uva.nl

Martha Larson
ISLA, University of Amsterdam
m.a.larson@uva.nl

Gareth J.F. Jones
Centre for Digital Video Processing
Dublin City University, Ireland
gjones@computing.dcu.ie

Maarten de Rijke
ISLA, University of Amsterdam
mdr@science.uva.nl

ABSTRACT

Spoken audio, like any time-continuous medium, is notoriously difficult to browse or skim without support of an interface providing semantically annotated jump points to signal the user where to listen in. Creation of time-aligned metadata by human annotators is prohibitively expensive, motivating the investigation of representations of segment-level semantic content based on transcripts generated by automatic speech recognition (ASR). This paper examines the feasibility of using term clouds to provide users with a structured representation of the semantic content of podcast episodes. Podcast episodes are visualized as a series of sub-episode segments, each represented by a term cloud derived from a transcript generated by automatic speech recognition (ASR). Quality of segment-level term clouds is measured quantitatively and their utility is investigated using a small-scale user study based on human labeled segment boundaries. Since the segment-level clouds generated from ASR-transcripts prove useful, we examine an adaptation of text tiling techniques to speech in order to be able to generate segments as part of a completely automated indexing and structuring system for browsing of spoken audio. Results demonstrate that the segments generated are comparable with human selected segment boundaries.

Categories and Subject Descriptors

H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing—*Indexing Methods*; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

General Terms

Algorithms, Experimentation

SSCS'08, July 24, 2008, Singapore.
.

Keywords

Speech browsing, term clouds, TextTiling

1. INTRODUCTION

Podcasting, the practice of publishing audio files online using a syndication feed, enjoys growing popularity. Users subscribe to podcasts or download individual episodes for listening. The podcasts available on the internet, known collectively as the podosphere, frequently contain unplanned conversational speech, including interviews, chitchat and user generated commentaries. The podosphere provides fertile ground for developing approaches to confront the difficulty presented by spontaneous speech content to automatic speech recognition (ASR) and automatic structuring algorithms. These include heterogeneous speaking styles, incomplete or interrupted sentences, lack of clear topical structure and unpredictable vocabularies. The growth rate of the podosphere makes providing access to the information contained in podcasts a significant and pressing speech search challenge.

Parallel to speech search on other domains, this challenge decomposes into two primary components. First, a user with an information need must be able to search the podosphere in order to locate podcasts or podcast episodes that satisfy that need. The task of podcast retrieval can be addressed with approaches that exploit podcast level metadata and ASR-transcripts, and the development of such techniques appears to be receiving substantial attention. Previous research demonstrates that speech retrieval is remarkably tolerant to high levels of transcription errors [5, 9, 17, 18, 23] and a growing number of podcast search engines on the internet, such as Podscope[1], Everyzing[2] and Pluggd[3] exploit a combination of speech-based and metadata features. Second, given a podcast

[1]http://www.podscope.com/
[2]http://search.everyzing.com/
[3]http://www.pluggd.tv/audio/

episode, the user must be able to easily preview that episode to confirm its relevance or else search within the episode to find specific sections that are relevant. The temporal nature of spoken data means that auditioning files to locate relevant information is often very time consuming and inefficient. Spoken audio differs in this way from text, which presents a user with an instantaneous impression accommodating depictions of importance or relevance in the form of highlighting or font changes such as size, boldness or coloration.

This aspect of podcast search is relatively less well studied, motivating us to pursue research on this issue. This paper devotes itself to the issue of representation of podcast episode content, investigating a method for creating articulated podcast episode surrogates that can be used by an interface to represent the semantic content of a podcast episode at the level of thematic segments. Previous work suggests that episode level surrogates are robust to speech recognition error [22] and this paper takes up the open question of whether this robustness is maintained at the segment level, where significantly fewer running transcript words are available to generate a given surrogate.

It is important to note that the issue of highly articulated semantic surrogates for podcast episodes is a fundamental one. Although improved ASR-transcripts would facilitate the generation of surrogates representing the semantic content of podcast episode segments, ASR-transcripts, even if they were completely error-free, are inherently unsuitable to serve as surrogates for speech content. Speaker prosody is an important component of spoken audio, modulating meaning and signaling structure. However, prosody fails to be represented in conventional ASR-transcripts. Unlike a newspaper report, a blog or an online product review, spoken content was not generated with the intent that it be interpreted in a purely textual form. The presence of repetition, false starts, and other disfluencies make transcripts of unplanned speech particularly difficult to skim. Finally, the sheer length of transcripts makes them unsuited for browsing or skimming and suggests that useful representations of podcast episodes would present semantic content to the user in abbreviated form. Moreover, even if podcast publishers provided highly detailed and accurate metadata describing podcast content, metadata alone would fail to provide a completely satisfactory surrogate, since it remains podcast level metadata; it describes the podcast as a whole and offers the user no more than a vague indication of where interesting points to listen in might be located. In the case where the user is looking for a particular audio clip, that is to say, a specific quote spoken by the speaker, detailed representations of local content within the episode are necessary.

In this paper we explore the effectiveness of the term cloud methods for the generation of structured representation of the semantic content of podcast episodes. Segment level term clouds are created from both reference transcripts and ASR-transcripts of a collection containing 30 podcast episodes on the topics of British history, music history and computer security downloaded from the internet. The quality of the segment clouds is compared using quantitative measures. The ability of segment clouds to enable users to identify a semantically specific section of a speech document is investigated with a small scale user study. Since the segment-level clouds generated from ASR-transcripts prove useful, we examine an adaptation of TextTiling techniques [12] to speech transcripts in order to be able to generate segments as part of a completely automated indexing and structuring system for browsing of spoken audio.

Developing this browsing approach poses a number of challenges. Suitable words which represent a segment must be extracted to form the term cloud, and the method used to do this must be optimised to minimise the impact of ASR transcription mistakes on the composition of term clouds. Key word redundancy compensates for ASR error for retrieval [9], so our hypothesis is that similar effects will extend to tag cloud creation. The objective is to have term clouds of sufficient quality so that users can reliably use them to identify relevant sections of a speech file, and that they should find the overall quality of the term clouds acceptable.

This paper examines the feasibility of using term clouds to provide users with a structured representation of the semantic content of podcast episodes. Podcast episodes are visualized as a series of sub-episode segments, each represented by a term cloud derived from a transcript generated by ASR. Results demonstrate that the segments generated are comparable with human selected segment boundaries.

This remainder is structured as follows. First, we provide a brief review of previous work in the areas of spoken content surrogates and in content segmentation. Then, we describe our method for generation of segment level term clouds and present results of evaluations of several methods for deriving these clouds from transcripts. Finally, we take a further step towards full automation of the process of automatic generation of topical segment level representations of podcast episodes by exploring the application of the TextTiling algorithm to ASR-transcripts. We finish with concluding remarks and an outlook on future work.

2. BACKGROUND

Issues of presenting speech documents to users and giving users a means by which to skim or browse within spoken content has attracted attention since research first began on speech search. Early examples include a system for interactive skimming [1] and an interface where keywords pop up upon roll-over of the audio segment containing them [4]. Visualizing spoken content is also a challenge that must be addressed in video search applications and has received attention since the beginnings of multimedia retrieval [3, 11]. The trend is movement towards finer-grained visualizations of spoken audio and methods of creating increasingly informative surrogates. Recent research has shown ASR-transcripts to be useful as the basis for generation of keyphrase summaries [6] and also snippet-like summaries [21]. These methods exploit redundancy in spoken content to generate noise robust spoken document representations and have motivated us to begin exploration into spoken audio surrogates that use *term clouds*. Term clouds are a weighted representation of significant words occurring in a document. They resemble tag clouds derived based on user tags of content [10] in that they contain words scaled visually according to importance. In previous work, we have shown that term clouds representing entire podcasts are resistant to noise from ASR-transcripts [22]. While this can assist a user in judging the relevance of a document to their information, the user is still required to listen to the complete document to find relevant information. Focused access to spoken audio motivates the investigation of techniques for topical structuring. Segmentation of spoken audio for purposes of information retrieval has been the subject of investigation since the 1990s with important examples including [8, 14, 15, 19, 20]. TextTiling is an algorithmic approach to automatically subdivide written text into semantically coherent segments that represent passages or subtopics. Patterns of lexical co-occurrence and distribution are used as triggers for identifying subtopic shifts. The concepts behind the TextTiling algorithm have already been applied to segmentation of other media such as audio [2, 16] and images [7].

Essentially TextTiling is designed to search for points in a running text where the vocabulary shifts from one topic to another. The TextTiling algorithm proceeds as follows:

- The transcription is broken into pseudo-sentences of size w referred to as *token sequences*.
- Token sequences are grouped together into groups of $blocksize$ to be compared against an adjacent group of token-sequences.
- The similarity between adjacent blocks b_1 and b_2 is then calculated using a cosine measure. This matching score is then assigned to the gap between the two blocks.

$$sim(b_1, b_2) = \frac{\sum_t w_{t,b_1} w_{t,b_2}}{\sqrt{\sum_t w_{t,b_1}^2 \sum_t w_{t,b_2}^2}}$$

where t ranges over all the terms from the tokenisation stage, w_{t,b_1} is the weight assigned to term t in block b_1. The weights are simply the frequency of terms within the block using the function.

$$w_{t,b_1} = \begin{cases} 1 + \log tf(t, b_i) & \text{if } tf(t, b_i) \geq 1 \\ 0 & \text{otherwise} \end{cases}$$

This produces a matching score in the range 0 to 1.

- Boundaries are determined by changes in the sequence of similarity scores. For a given token-sequence gap i, the algorithm looks at the scores of the token-sequence gaps to the left of i as long as their values are increasing. When the peak is reached the difference between the score at the peak and the score at i is recorded. The same procedure takes place with the token-sequence gaps to the right of i. The relative height of the peak to the right of i is added to the relative height of the peak to the left. A gap occurring at a peak has a score of zero since neither of its neighbours is higher than it. These new scores, called *depth scores*, are then sorted. Segment boundaries are assigned to the token-sequence gaps with the largest corresponding scores.

In the original text-based algorithm segment boundaries are then adjusted to correspond to true paragraph breaks. Since ASR transcripts lack sentence and paragraph ends it is impossible to exploit these natural segmentation points. To compensate for this, the intention is to use timing information and low energy points from the ASR to adjust the boundary position of a detected segment. Low energy points could be thought as pauses in discourse, which in turn could be interpreted as paragraph breaks [16].

Segmentation is a complex process where even human segmenters can on occasion disagree on appropriate segmentation points. TextTiling is thus likely to be imperfect for this task. We wish to determine whether it is sufficiently accurate for our purposes of segment-based term cloud generation, and to what extent errors in ASR transcriptions affect its performance.

3. DATA SET

In this section we describe the data set that we use for our experiments involving the segment-level term clouds and automatic segmentation.

3.1 Podcasts

The data used for our research were 30 podcast episodes, 10 each from three U.S. English language podcasts: 10 episodes from Security Now [4] (SN) (9h 7m), 10 from British History [5] (BH) 101 (2h 7m) and 10 from Music History Podcast [6] (MH) (2h 48m). SN contains spontaneous conversational speech of two persons, BH and MH are narratives of a single speaker. As the titles may suggest, SN content focuses on the computer security domain, BH on British history and MH on music history. The recordings have been made by private individuals, who we assume do not make use of studio facilities.

The speech recogniser generated transcripts with word error rates that are higher than they would have been for dictated speech. No word error rate was calculated but the annotator noted that the broad vocabularies of MH and BH and the conversational style of SN posed a challenge to the recogniser.

3.2 Podcast Episode Transcripts

The three podcasts in our data set are particularly interesting for experimental investigations since their publishers make transcripts available along with the audio file.

The transcripts were used as the reference transcripts in our experiments and we estimate them to have approximately a 95% word accuracy rate. The ASR transcripts used in the experiments were generated using Nuance Dragon Naturally Speaking SDK Server Edition[7] directly out of the box. No language model adaptation or speaker profiles were used, only the default model for U.S. English.

Both ASR and human transcripts were post-processed in order that they could be aligned. We removed descriptive information not contained in the spoken content, such as title and copyright information, from every SN episode. In MH, the speaker deviates from the transcript by adding to the introduction and epilogue. To accomodate this we had to remove the additional text introduced in ASR output.

The transcriptions are further processed using Porter's stemmer to conflate varied word forms and stop word removal[8] was applied to remove noise to prevent high frequency function words from appearing in term clouds and to help the segmentation borders become clearer.

3.3 Reference Segmentation of Transcripts

The podcast episodes in the corpus were manually segmented by one human annotator according to topic shifts in the content. The segment boundaries were first marked by the annotator in the reference transcripts and subsequently the corresponding segment boundaries were identified in the ASR transcripts. At the same time material not present in the ASR transcripts was removed, as is discussed further below. The purpose of the manual annotation was to establish a gold standard to be used first for investigation of segment-level term cloud generation and then for evaluation of the automatic segmentation.

Determination of topical segment boundaries in podcast episodes requires delicate judgments concerning topicality. Most podcast episodes focus one main topic and for this reason topic shifts are rather subtle and the segments mirror different aspects of the episode's main topic rather than fundamentally different topics. This makes it difficult to develop strict and clear rules for topic segmentation that are still flexible enough to be applicable to all different kinds of podcasts in the corpus. However, the following rules of thumb were used for the segmentation of the human transcripts: Segment boundaries were marked after the prologue and before the epilogue of an episode, if there were any. These parts often contain introductory or summarizing information about the episode without contributing to its actual content. Also sections that contained adver-

[4] http://www.grc.com/securitynow.htm
[5] http://bh101.wordpress.com
[6] http://www.musichistorypodcast.com
[7] http://www.nuance.com/audiomining/sdk
[8] http://staff.science.uva.nl/∼tsagias/sscs.htm

tisements or other meta-level information were marked as distinct segments. Beyond these rules, it was left to the annotator to decide on appropriate points for segment shifts, aiming at a fine-grained segmentation of topics. The annotator was asked to mark segment boundaries at points that would be good entry points for a listener switching between the segments. The ASR transcripts were segmented by identifying the segment shifts in the human transcripts and finding the corresponding points in the ASR transcripts.

Some issues we encountered in connection with topic segmentation were the following:

- The manual transcripts constitute plain versions of the episodes' content, meaning that they were largely cleaned of the typical characteristics of spoken language such as unintended repetition of words and similar phenomena. In many cases what was earlier referred to as meta-level information, e.g., advertisements, was also excluded from the transcripts. In contrast, the speech recognizer produced transcripts for the full speech content of the episodes. This introduced some cases in which the content of the human and the ASR transcripts did not correspond. Large differences in the transcripts make the data noisy which makes it necessary to handle such cases carefully.
- An issue related to ASR errors was that the exact point of the segment shift, as identified in the human transcript, was not always identifiable in the ASR transcripts. However, we believe that the minor differences in segment boundaries that result from this are small or no impact for the content of the topic segments because topics are generally long enough to tolerate such variance.

4. SEGMENT-LEVEL TERM CLOUDS

In this section we describe how we create segment-level term clouds. We evaluate segment-level clouds created on the basis of a segmentation of podcast transcripts created by the human annotator. We present the results of our evaluation, both using quantitative measures and using a small scale user study.

4.1 Term Cloud Generation

For each podcast episode in our data set, we generated visual surrogates consisting of a series of term clouds, one for each segment. The segment boundaries used were the reference segment boundaries generated by the human annotator. The term cloud for each segment was based on the transcript of each segment. We refer to segment level term clouds generated from the reference (publisher provided) transcripts as *reference clouds* and the segment level term clouds generated from the ASR transcripts as *ASR clouds*. Common practice for tag cloud generation (flickr, del.ici.ous, technorati, blinklist, blogmarks, simpy) uses alphabetical term ordering. The size of font used scales linearly with the weight of a term. We adopted these conventions for the visualization of the clouds that we used in our experiments and analysis.

We investigated two different term weighting techniques: a Term Frequency (tf) only method and a method combining Term Frequency and Inverse Segment Frequency ($tf.isf$). To calculate tf we counted the number of occurrences of the stem form of each word in a given segment. To calculate isf we take the inverse of the number of segments within the podcast episode that contain a particular term. Our isf is thus a parallel of the standard calculation of inverse document frequency idf.

It was observed that segment term clouds of an episode produced only using tf served to distinguish a podcast episode segments from segments of other episodes within the podcast or within the collection. However, tf-based term clouds didn't appear to offer maximum discrimination of semantic content between the segments of a single podcast. In other words, it is easy to distinguish between a pair of tf-based term clouds if they are from different episodes, but not if they are from different segments within the same episode.

We also explored $tf.isf$-based clouds in order to be able to experiment with clouds that represent segment-level semantic content, but do it in a way that allows distinction among segments contained in a single episode.

In our investigation we experiment with different term cloud sizes. Specifically, we analyzed two fixed size clouds of 5 and 10 terms. Additionally, we observed that podcast episode segments differ greatly with respect to length. This difference suggested that a fixed sized cloud may not be appropriate for representation of all segments, since longer segments could potentially (but not necessarily) contain a greater number of semantically important terms. We created dynamically-sized clouds in which the number of terms was set at 10% of the number of words contained in the transcript of the segment (after stopword removal). The variable-sized clouds were chosen for further investigation in the user studies.

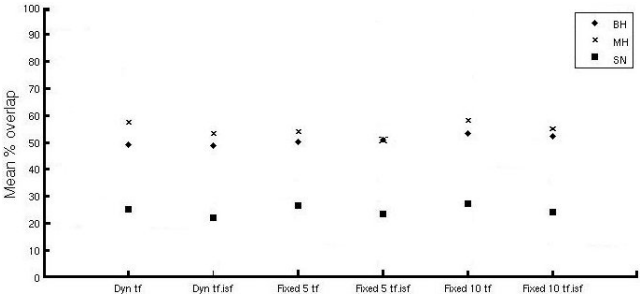

Figure 1: Graph of mean overlap of ASR-cloud and reference-cloud vocabularies for all experimental conditions

Table 1: Mean overlap of tf-based clouds generated from human and ASR-transcripts

	Dynamic	Fixed 5	Fixed 10
BH	0.48 ± 0.31	0.49 ± 0.24	0.53 ± 0.22
MH	0.57 ± 0.24	0.55 ± 0.25	0.59 ± 0.18
SN	0.25 ± 0.32	0.27 ± 0.31	0.27 ± 0.29
Average	0.43	0.44	0.46

Table 2: Mean overlap of $tf \cdot isf$-based clouds generated from human and ASR-transcripts

	Dynamic	Fixed 5	Fixed 10
BH	0.48 ± 0.33	0.50 ± 0.27	0.52 ± 0.24
MH	0.54 ± 0.25	0.51 ± 0.22	0.56 ± 0.18
SN	0.22 ± 0.30	0.24 ± 0.31	0.24 ± 0.29
Average	0.41	0.42	0.44

4.2 Quantitative Analysis of Term Clouds

Our quantitative analysis of segment-level term clouds is carried out with the same metrics we previously applied in [22]. We take the clouds generated from the reference transcripts as the standard for a perfect term cloud and compare these clouds to clouds generated for the same segments using the content of the ASR-transcripts. First, we apply a set-based measure of term cloud quality. This measure, called Term Overlap, is the proportion of

Table 3: Mean Spearman correlation coefficient and p-values using tf-based clouds generated from human and ASR-transcripts

	Dynamic		Fixed 5		Fixed 10	
	Coef	p-val	Coef	p-val	Coef	p-val
BH	0.49	0.43	0.60	0.60	0.73	0.21
MH	0.68	0.39	0.60	0.59	0.74	0.22
SN	0.29	0.18	0.27	0.27	0.41	0.16
Average	0.49		0.49		0.63	

Table 4: Mean Spearman correlation coefficient and p-values using $tf \cdot isf$-based clouds generated from human and ASR-transcripts

	Dynamic		Fixed 5		Fixed 10	
	Coef	p-val	Coef	p-val	Coef	p-val
BH	0.58	0.39	0.54	0.57	0.62	0.24
MH	0.62	0.42	0.51	0.66	0.70	0.29
SN	0.23	0.17	0.29	0.24	0.31	0.14
Average	0.47		0.45		0.54	

the number of terms that overlap between the vocabularies of two clouds. In order to make the clouds comparable, the vocabulary of the larger cloud is truncated to the size of the vocabulary of the smaller cloud. This truncation is performed on the list of the terms in the clouds vocabulary ranked by weight. Inequalities in cloud size come about because segments covering the same content can contain a different number of words in the speech transcripts than are present in the reference transcripts.

Table 1 reports overlap proportions for tf-based clouds and Table 2 for $tf.isf$-based clouds. It can be seen that in both cases, ASR clouds show healthy overlap with reference clouds. The overlap is relatively independent of whether a cloud is 5 or 10 terms large or dynamically generated. The vocabularies of the reference clouds and the ASR clouds have the least overlap for the Computer Security (SN) podcast. We believe that the low overlap is related to the very small segment size arising from the fact that SN involves often rapid fire conversation. We conjecture that the segment transcripts simply do not contain enough words to demonstrate the compensation for ASR-error noise enjoyed by clouds generated from larger sections of transcript. For comparison purposes, all overlaps are summarized in Figure 1.

Next, we investigate the relationship of term weights between reference clouds and ASR clouds by calculating the Spearman's rank correlation coefficient between the frequency-ranked list of the reference term cloud vocabulary and the ASR term cloud vocabulary. These results are reported in Table 3 for tf-based clouds and in Table 4 for $tf.isf$-based clouds. These results suggest that the weights assigned to terms in ASR-clouds are comparable to weights assigned to terms in the reference clouds.

According to our measures, segment-level clouds generated from ASR transcripts do not attain the quality of episode level clouds. In previous work [22] we measured overlaps of around 0.70 for podcast episode level clouds on the same corpus. We believe, however, that the segment level speech clouds are of sufficient quality to support users in skimming podcast episodes and in selecting points at which to begin listening. We investigated this hypothesis by performing a small scale user study which is reported in the next section.

4.3 User Study of Segment-level Clouds

In these user studies we examine the effectiveness of term clouds to enable users to locate relevant information within a group of topically cohesive sub-document segments. We want to determine how well users are able to use segment-level clouds to identify document segments containing particular semantic content. A group of 10 individuals, mostly postgraduate students, participated in the study. Each subject carried out 30 tasks. A task consisted of a sentence extracted from one of the podcast episode segments and five segment-level term clouds, one of which was generated from the transcript of the segment containing the sentence. The other four clouds were created from other segments from the same podcast episode. One task was created for each of the 30 podcast episodes in the experimental data set by randomly selecting a segment from that podcast and choosing a sentence that was highly representative of the content of that segment. In this way, we ensured that the sentence presented to the user was representative of an information need optimally satisfied by the segment containing it. For each task, subjects were asked to rank the five term clouds in the order in which they felt the term clouds best matched the given sentence. Out of the 30 tasks, the first 3 were allocated as practice tasks to allow the subject to become acquainted with the task. The time taken for each user to complete each task was also recorded. Effectively the user study involved a total of 60 tasks, since each task had a reference-cloud version in which all 5 clouds were derived from the reference transcripts and an ASR-cloud version in which all 5 clouds were derived from the ASR transcripts. Since ASR transcripts can be subject to errors due to the speech recognition process and thus may contain incoherent or nonsensical passages, the sentence chosen to represent a segment in the ASR-cloud version of a task drawn from the reference transcript, in other words, was the same sentence used in the reference-cloud version of the task. In designing the task, we emulated a use case in which a user would want to pick out the segment closest to a particular information need from among segments that were semantically related, but not optimal matches, i.e, we investigated the case in which a user is interested in exploiting the ability of a segment-level cloud to provide semantic discrimination between a particular topical segment and other topical segments in the same podcast episode. In order to ensure that the discriminative decision was a realistically challenging one, the alternate clouds presented to the user were those whose segments were "closest" in content to the segment containing the task sentence. We determined "closest" by indexing the data set using the Lucene [9] search engine and retrieving the top five most relevant clouds for a given sentence.

In order to randomly assign a well-balanced mix of ASR clouds and reference clouds derived from different podcasts to each subject, the 60 transcripts in the reference data set were divided into two smaller data sets, A and B. Each contained 30 tasks: 15 ASR cloud tasks and 15 reference cloud tasks. The ASR transcripts in A corresponded to the Human transcripts in B, the Human transcripts in A corresponded to the ASR in B and vice versa. Five users completed the user study with data set A while the other five completed the study with data set B. The term clouds generated for the user study were sized proportionately to the number of words in the segment. First, the stopwords were removed, then the cloud size was set at 10% of the number of running words contained in a segment.

The results of the user study are shown in Figures 2 and 3, which shows the correlation between the choices made by users during the user studies and the correct term clouds. The lighter bar on the left depicts the reference cloud version of each task and the darker bar on the right depicts the ASR cloud version of the task. Recall that no single subject performed both versions of the same task, but

[9] http://lucene.apache.org/

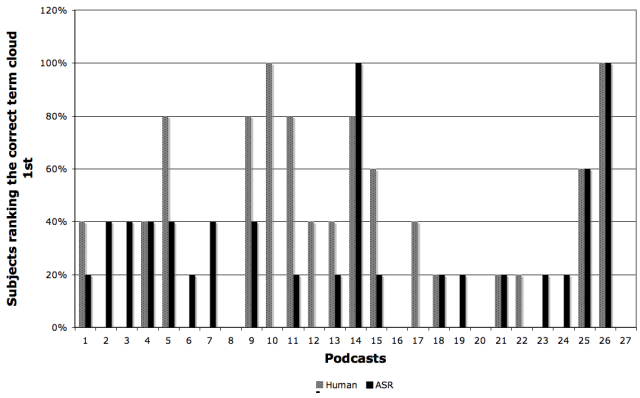

Figure 2: Results from user studies, showing correlation between the top subject cloud choices for reference clouds (grey on left) and ASR-clouds (black on right)

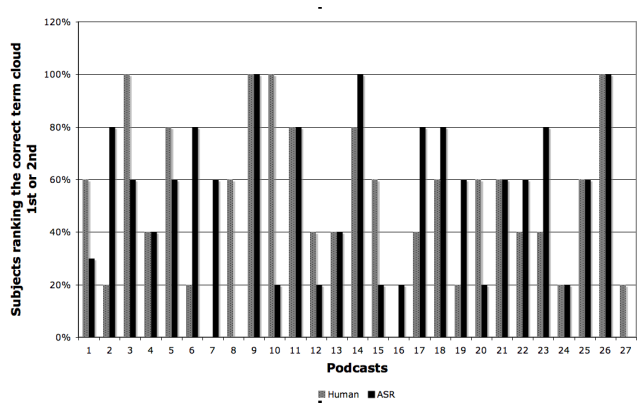

Figure 3: Results from user studies, showing correlation between top two subject choices for reference clouds (grey on left) and ASR-clouds (black on right)

that the tasks were distributed over subjects in a carefully balanced manner. Figure 2 shows how often subjects selected the term cloud representing the segment containing the task sentence as their top choice. While there is some correlation evident, it is not particularly strong. We note that within the skimming/browsing use case that interests us, the user is most likely to invest more than one click in finding a relevant segment. For this reason, we also investigated the correlation between the subjects' top two choices and the correct term cloud. In this case, shown in Figure 3, we see a much stronger link between subject performance on the reference cloud version of the tasks and on the ASR cloud version of the tasks. The experimental results suggest that segment level clouds provide users with surrogates that are adequate representations of segment level semantic content. Additionally, the results indicate that at the segment level ASR clouds are comparable with reference clouds with respect to their ability to support a user in a discriminative choice of a semantically relevant segment made about podcast segments occurring in the same episode.

Table 5 shows the average amount of time taken by users per task. We can see that for the BH and SN parts of the set, users took slightly longer to complete the task when dealing with the ASR transcripts. This is in line with our expectations (i.e., that any noise from the ASR transcription will have a detrimental effect on performance times). However, in the case of the MH domain, the opposite is true, with users taking up to 10 seconds longer to perform the task on manual transcripts. We suggest two possible causes for this: firstly, that due to the high similarity of the suggested term clouds, users need to take longer to distinguish between them; secondly, it could be caused by the ASR performing particularly badly on this domain (musical history) resulted in very poor term clouds, to which the users simply assigned a preference at random, unable to distinguish any qualitative difference. These problems may be overcome by improving the vocabulary coverage of the speech recogniser system, or by performing a keyword search on the audio to look for specific terms which may be mined from alternative sources such as podcast metadata or related manual transcripts.

Table 5: Average time per test case

Time	MH	BH	SN	Overall
ASR	42.33	40.66	38.11	40.3
Manual	52.95	37.84	36.18	41.8

So after listening to me talk about all the German innovations to the way the instrument was played and built, why do we not call it the German horn? Well, actually the U.S., Britain and Canada are the only countries to really refer to this instrument as the French Horn. Most of Europe has always has and still does refer to it as simply the horn or horn in F. Nobody knows for sure why it became known as the French horn. The distinction was probably made back in its days as being primarily used as a hunting instrument. It could be that it was in France that the Germans got the idea from when the technique was brought to them by Count Franz Anton von Sporck. **Another more popular possibility is that the British and French hunting horns differed in size, and the size of the horn eventually used in the orchestra reminded the British of the hunting horns used by the French.**

Figure 4: An example segment from a podcast episode with the user task sentence marked in bold

The user study demonstrated that subjects often reject the correct term cloud in favour of a similar alternative. Examination of the data turned up a possible explanation for this pattern. We noticed that in many of the cases where the correct term cloud was rejected that term cloud often contained a significant, highly specific term which was not present in the representative sentence of the passage. We conjecture that the presence of this term causes the user to reject the term cloud as a whole, as the term seems unrelated to the presented data. We illustrate this with an example in Figure 4 which depicts the segment with the task sentence marked in bold. The term cloud most frequently chosen for this passage contained the terms *cor, count, horn, hunting, le, means, played* whereas the correct term cloud contained the words *french, german, horn, hunting*. We believe that since the term "German" occurs in the term cloud, yet not in the sentence, which is representative of the user information need, experimental subjects were led to reject this term cloud. Since it was juxtaposed with terms such as "British" and "French" occurring in the task sentence, subjects apparently decided that the term cloud was not representative of the passage, and thus tended to choose the more generic alternative.

5. SPEECH TILING EXPERIMENTS

Table 6: Results of TextTiling algorithm applied to ASR-transcripts

ASRref	Precision	Recall	F1	no. Ref. Segs
MH	0.69	0.88	0.76	5.8
BH	0.82	0.62	0.7	10.3
SN	0.61	0.8	0.7	25
Average	0.71	0.77	0.72	13.7

Table 7: Results of TextTiling algorithm applied to manual transcripts

HRef	Precision	Recall	F1	no. Ref. Segs
MH	0.69	0.89	0.77	5.8
BH	0.9	0.67	0.76	10.3
SN	0.517	0.872	0.65	23.5
Average	0.70	0.81	0.73	13.2

In the previous section reporting our evaluation of segment-level term clouds, we have demonstrated that segment-level clouds have the potential to provide users with valuable support in identifying and discriminating podcast episode segments according to their semantic content. We also have concluded that segment clouds generated from ASR transcripts are comparable to segment clouds generated from reference transcripts. In terms of automating the process of creating articulated surrogates of podcast episodes for browsing and skimming spoken audio this result is significant and leads naturally to a new task of finding a means of automatically setting boundaries of topical segments in ASR transcipts.

The technique of TextTiling was chosen for our exploratory experiments into automatic generation of topical segmentations of podcast episodes using ASR transcripts. We evaluate the quality of the segments produced by the algorithm by comparing the segment boundaries with the segment boundaries set by the annotator who produced the reference segmentation described above. Table 6 reports the values of precision and recall and also the F1-value for this evaluation. Precision and recall were calculated by aligning the automatically produced segment boundaries with the reference segment boundaries. Each automatically generated boundary point was aligned with the closest occurring reference boundary points. Boundary points that fell between already aligned boundary points couldn't be matched to any reference points and so were discarded. Recall is calculated as the proportion of reference boundaries that could be aligned to automatic boundaries and precision is calculated as the proportion of automatic boundaries which could be aligned to reference boundaries. In order to give an impression of the nature of the task, the average number of segments in the reference transcripts is also reported. In essence, the TextTiling algorithm produces segments which agree quite well with those produced by the manual process. Further experimentation with speech tiling is required to improve these results and conduct studies with human evaluators.

Table 7 reports the results from the same TextTiling experiments, but applied to manually produced transcripts rather than ASR transcripts. We show here that similar results are obtained by TextTiling, regardless of the source material used.

6. CONCLUSION AND OUTLOOK

This paper has presented a new method for representing the semantic content of podcast episodes in a way that helps users to skim a podcast for its content and to determine appropriate listening in points. Podcast episodes were represented as a series of term clouds derived from ASR transcripts, one cloud per topical segment. Results of our study show that segment-level term clouds generated from ASR transcripts are comparable to those generated from reference transcripts in their ability to convey semantic information relating to the contents of the source material. These results motivated an investigation of the application of TextTiling techniques to ASR transcripts in order to automatically generate the boundaries of the segments on which the segment-level term clouds are based. Evaluation of this approach showed that automatic techniques generate boundaries very close to those chosen by a human annotator. These results suggest that it is possible to fully automate the process of transcribing, indexing and searching podcasts.

While the results of this initial study are encouraging, there are many areas of further work that can be carried out to improve and extend this approach. We would like to more fully explore the possibilities for selecting words to include in the term clouds. Our experiments were carried out using term clouds generated exclusively from ASR transcripts generated by a speech recognizer not trained to the domain. In the future, we would like to investigate exploiting the metadata (in particular, the contents of the title and the description elements) included in the podcast feed in order to adapt the language model of the speech recognizer or else to directly influence the choice of words and weights for inclusion in the cloud. Developing techniques for use of metadata would necessarily include compensating for incomplete, inaccurate or missing metadata, mentioned above as being problematic in the podcast domain.

Future investigation is needed to answer the question if longer segments must indeed be represented with larger term clouds. It is possible that longer segments contain more semantically critical terms, but this needs to be supported by empirical studies. Another factor that we feel is important in determining the size of the segment-based cloud is whether the user expects the size of the cloud to represent the length of the segment. If so, term cloud size may actually be important in conveying information about segment length and may be less dependent on the number of semantically important terms that are spoken during a segment.

Additionally, we would like to more fully explore the possibilities of how to present term clouds to the user. Currently we present them in alphabetical order which is the conventional approach for user annotated tag clouds, but term clouds extracted from documents could possibly be more effective if words were presented in the order in which they occur in the original document or if some method of term weighting was employed [13].

Another area that we have identified for future work is the refinement of the adaptation of the TextTiling technique to speech transcripts. Content segmentation is currently based only on the document transcription, but potentially non-verbal features such as silence points, boundaries between speech and music and speaker change points could be useful to improve the segmentation accuracy.

Finally, we would like to directly investigate segment-level term clouds created from automatically generated podcast episode segments. In the work presented here, we tested clouds generated using segment boundary points set by the human annotator. However, the quality of automatically generated segment boundary points is such that suggests that they will also support generation of segment-level clouds of adequate quality to be beneficial to the user. The parameters of the TextTiling algorithm make it possible to generate segmentations with varying levels of fineness. We anticipate that there is an optimum operating point at which segments are small enough to yield a series of clouds that represents the full articulation of the semantic content of the document, but large enough to ensure that the segment-level clouds retain the necessary robustness

to speech recognition error.

This paper has provided an initial demonstration of the usefulness of term clouds for representing the semantic content of podcasts on a topical segment level and has opened a series of vistas on future development of segment-level cloud surrogates for spoken content.

7. ACKNOWLEDGEMENTS

This research was supported by the Netherlands Organization for Scientific Research (NWO) under project numbers 220-80-001, 017.001.190, 640.001.501, 640.002.501, 612.066.512, and by the Dutch-Flemish research programme STEVIN under project DuOMAn (STE-09-12), and by the E.U. IST programme of the 6th FP for RTD under project MultiMATCH contract IST-033104.

References

[1] B. Arons. Speechskimmer: a system for interactively skimming recorded speech. *ACM Trans. Comput.-Hum. Interact.*, 4(1):3–38, 1997. ISSN 1073-0516. doi: http://doi.acm.org/10.1145/244754.244758.

[2] S. Banerjee and A. Rudnicky. A texttiling based approach to topic boundary detection in meetings. In *INTERSPEECH-2006*, 2006.

[3] M. G. Brown, J. T. Foote, G. J. F. Jones, K. S. Jones, and S. J. Young. Automatic content-based retrieval of broadcast news. In *ACM Multimedia*, pages 35–43, 1995.

[4] M. G. Brown, J. T. Foote, G. J. F. Jones, K. S. Jones, and S. J. Young. Open-vocabulary speech indexing for voice and video mail retrieval. In *MULTIMEDIA '96: Proceedings of the fourth ACM international conference on Multimedia*, pages 307–316, New York, NY, USA, 1996. ACM. ISBN 0-89791-871-1. doi: http://doi.acm.org/10.1145/244130.244232.

[5] W. Byrne et al. Automatic recognition of spontaneous speech for access to multilingual oral history archives. *IEEE Transactions on Speech and Audio Processing, Special Issue on Spontaneous Speech Processing*, 12(4):420–435, 2004.

[6] A. Désilets, B. de Bruijn, and J. Martin. Extracting keyphrases from spoken audio documents. In *Information Retrieval Techniques for Speech Applications*, pages 36–50, London, UK, 2002. Springer.

[7] A. Doherty and A. F. Smeaton. Automatically segmenting lifelog data into events. In *WIAMIS 2008 - 9th International Workshop on Image Analysis for Multimedia Interactive Services*, 2008.

[8] M. Franz and J.-M. Xu. Story segmentation of broadcast news in arabic, chinese and english using multi-window features. In *SIGIR '07: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 703–704, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-597-7.

[9] J. Garofolo, G. Auzanne, and E. Voorhees. The trec spoken document retrieval track: A success story. In *6th RIAO Conference: Content-Based Multimedia Information Access*, pages 1–20, April 2000.

[10] Y. Hassan-Montero and V. Herrero-Solana. Improving tag-clouds as visual information retrieval interfaces. In *International Conference on Multidisciplinary Information Sciences and Technologies (InSciT2006)*, Mèrida, Spain, 2006.

[11] A. G. Hauptmann and M. J. Witbrock. Informedia: news-on-demand multimedia information acquisition and retrieval. pages 215–239, 1997.

[12] M. A. Hearst. Texttiling: segmenting text into multi-paragraph subtopic passages. *Comput. Linguist.*, 23(1):33–64, 1997. ISSN 0891-2017.

[13] M. A. Hearst and D. Rosner. Tag clouds: Data analysis tool or social signaller? In *Proceedings of 41st Hawaii International Conference on System Sciences (HICSS 2008), Social Spaces minitrack*, 2008.

[14] P.-Y. Hsueh. Audio-based unsupervised segmentation of multiparty dialogue. ICASSP, 2008.

[15] P.-Y. Hsueh and J. D. Moore. Automatic topic segmentation and labeling in multiparty dialogue. In *Spoken Language Technology Workshop, 2006. IEEE*, pages 98–101. IEEE, December 2006. ISBN 1-4244-0873-3.

[16] G. J. F. Jones and R. J. Edens. Automated alignment and annotation of audio-visual presentations. In *ECDL '02: Proceedings of the 6th European Conference on Research and Advanced Technology for Digital Libraries*, pages 276–291, London, UK, 2002. Springer-Verlag. ISBN 3-540-44178-6.

[17] K. Koumpis and S. Renals. Content-based access to spoken audio. *Signal Processing Magazine, IEEE*, 22(5):61–69, 2005.

[18] P. Pecina, P. Hoffmannova, G. J. F. Jones, Y. Zhang, and D. W. Oard. Overview of the clef 2007 cross-language speech retrieval track. In *Proceedings of the CLEF 2007: Workshop on Cross-Language Information Retrieval and Evaluation*, Budapest, Hungary, 2007.

[19] T. Robinson, D. Abberley, D. Kirby, and S. Renals. Recognition, indexing and retrieval of british broadcast news with the THISL system. In *Proceedings of Eurospeech'99*, 1999.

[20] A. Rosenberg and J. Hirschberg. Story segmentation of broadcast news in english, mandarin and arabic. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 125–128, New York City, USA, June 2006. Association for Computational Linguistics.

[21] X. Shou, M. Sanderson, and N. Tuffs. The relationship of word error rate to document ranking. In *AAAI Spring Symposium, Technical Report SS-03-08*, pages 28–33, 2003.

[22] M. Tsagias, M. Larson, and M. de Rijke. Term clouds as surrogates for user generated speech. In *Proceedings of SIGIR 2008*, 2008.

[23] M. J. Witbrock and A. G. Hauptmann. Speech recognition and information retrieval: Experiments in retrieving spoken documents. In *Proceedings of the DARPA Speech Recognition Workshop*, pages 160–164, Virginia, U.S.A., 1997.