Experiments with Segmentation Strategies for Passage Retrieval in Audio-Visual Documents

Petra Galuščáková and Pavel Pecina
Charles University in Prague
Faculty of Mathematics and Physics
Institute of Formal and Applied Linguistics
Prague, Czech Republic
{galuscakova,pecina}@ufal.mff.cuni.cz

ABSTRACT

This paper deals with Information Retrieval from audio-visual recordings. Such recordings are often quite long and users may want to find the exact starting points of relevant passages they search for. In Passage Retrieval, the recordings are automatically segmented into smaller parts, on which the standard retrieval techniques are applied. In this paper, we discuss various techniques for segmentation of audio-visual recordings and focus on machine learning approaches which decide on segment boundaries based on various features combined in a decision-tree model. Our experiments are carried out on the data used for the Search and Hyperlinking Task and Similar Segments in Social Speech Task of the MediaEval Benchmark 2013.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Search process, Selection process*

General Terms

Experimentation, Theory

Keywords

Speech retrieval, Passage retrieval, Semantic segmentation

1. INTRODUCTION

Information Retrieval (IR) is a task which involves searching for documents relevant to a given query. In standard IR, the documents are usually in written form. In this work, we focus on retrieval from audio-visual documents (recordings). This task is even more demanding than the traditional IR from text documents because the semantic content of recording needs to be mined using an Automatic Speech Recognition (ASR) system applied to the audio track or using video content analysis of the visual track. The recordings have a linear structure and, compared to texts, are harder to skim, which causes a problem especially for long recordings. In our approach, we apply Passage Retrieval to audio-visual recordings – a process which splits long documents into series of shorter passages which serve as individual documents for the subsequent IR setup. This enables users to find the exact relevant segments in a collection of long audio-visual documents and should reduce the time required to find the requested information. In addition, text documents are often structured (into title, sections, paragraphs, etc.) by their authors, however, no such structure is usually given in audio-visual recordings. To some extent, the structure of recordings can be derived from audio and visual features (such as shot change or length of silence), but only with a limited reliability.

In the following section, we introduce Passage Retrieval in general. In Section 3, we discuss segmentation into semantically coherent passages, which is the main focus of this work. In Section 4, we describe data and evaluation methods applied in our experiments. Our experimental setup and results of Passage Retrieval based on different segmentation strategies applied to the MediaEval Benchmark data are described in Section 5. Section 6 then concludes the paper and gives directions for future work.

ICMR '14, Apr 01-04 2014, Glasgow, United Kingdom
ACM 978-1-4503-2782-4/14/04.
http://dx.doi.org/10.1145/2578726.2578753.

2. PASSAGE RETRIEVAL

Information Retrieval is the process of searching through a collection of documents, finding those which are relevant to a user's query, and returning full documents as a result. However, such retrieval of full documents might be, in some cases, insufficient. Passage Retrieval splits texts into smaller units which then function as documents in the information retrieval process and make it more precise.

Passage Retrieval techniques were shown to help also in the "classical" IR. First, the positional information of term occurrences (usually ignored in IR) can be used in indexing and term weighting [19], e.g., by assigning higher weights to terms occurring near the beginning of documents. Second, Passage Retrieval can improve results of IR for long documents containing a large range of different topics. If a document contains a short relevant passage among many other (irrelevant) passages, the document is often (incorrectly) identified as not relevant. In Passage Retrieval, the searched terms must appear within a limited distance which can subsequently improve retrieval performance of full documents too (see, e.g., [22, 12]). Third, document length normalization (frequently used in IR) can be realized based

on the length of the detected segments and not the entire documents. Kaszkiel and Zobel [12] show that this approach is very useful, especially for similarity measures which tend to prefer shorter documents (e.g., cosine). The final advantage of Passage Retrieval in IR is that identification of exact positions of relevant passages in long documents reduces users' time required to find the requested piece of information. This is even more important for long audio-visual recordings, where skimming is more time-demanding.

Segmentation for audio-visual recordings has not been widely studied; some experiments were performed by Eskevich et al. [7] and Wartena [31]. The Story Segmentation Task of TRECVID 2003 [24] and 2004 [25] focused on identification of story boundaries in video recordings, however, the detected boundaries were not subsequently used for IR. By contrast, in the Search Task of TRECVID, the goal was to retrieve shots relevant to given topics, but their boundaries were given on the input.

Kaszkiel and Zobel [13] divide segmentation strategies for Passage Retrieval into three groups – window-based (passages are created regularly as overlapping windows of fixed length, measured in terms of words), structure-based (defined by the author of the document), and semantic-based (corresponds to the real topical structure of documents).

Surprisingly, in text retrieval, the majority of authors (e.g., [31, 3, 13, 28]) show that the segmentation using sliding windows and creating overlapping segments of a regular length is the most successful approach to segmentation and its subsequent usage in IR. It is also demonstrated that this approach is sensitive to the window length which needs to be tuned on training data. For instance, Callan [3] uses a window of about 200–250 words, similarly Kaszkiel and Zobel [13] achieve the best results with 150–300 words, and Wartena [31] achieves the best result on audio-visual recordings with about 20 content words.

The window-based approach also achieves the best results in audio-visual retrieval experiments by Eskevich et al. [7] who compare segmentation techniques of the Rich Speech Retrieval Track participants in MediaEval Benchmarking in 2011. Wartena [31] compares four segmentation approaches and evaluates the segmentation quality of audio-visual data. The author also concludes that the quality of retrieval is sensitive to segment length. The best result is achieved using a sliding window but the segmentation into topically coherent segments proves to be more robust and less sensitive to the predefined average length of the segment. It achieves better results for longer segments and thus enables reduction of the total number of segments.

A possible explanation of the poorer results of structure-based segmentation is that it produces segments of highly variable lengths [13]. According to Tiedemann and Mur [28], the actual segmentation method is not crucial; it is the length of the segments what is more important in this task. Their semantic-based segmentation (based on coreference chains and the TextTiling [9] algorithm) outperforms both: segmentation which is based on paragraphs and sections defined by the author and regular segmentation. They illustrate this improvement on the task of question answering.

3. SEMANTIC SEGMENTATION

The goal in this work is to improve IR from audio-visual recordings with no or only weakly specified predefined structure. We employ segmentation methods originally designed for textual documents, here applied to ASR transcripts, combined with other features (e.g., audio) automatically extracted from the recordings. By using semantic segmentation, we are able to control the length and the nature of the segments. Therefore, semantic segmentation could be an effective method for splitting audio-visual documents for IR. This research area is quite new and only a limited number of works have been published on this topic [7, 31].

In this section, we discuss algorithms for semantic segmentation which aim at detection of passages which are semantically coherent and cover single topics. Segmentation should be consistent with the final task and correspond to expected answers to user queries. Segmentation may also vary according to the type of the data collection. In TV news programmes, the segments should correspond to the individual stories or their parts and thus they should be relatively clearly separable. On the other hand, topic boundaries in casual conversation or movies will be more blurred.

In general, the segmentation methods can be divided into similarity-based, lexical-chain-based, and feature-based [18, 14]. In the following subsections, we describe these approaches with respect to the modality of the input data.

3.1 Text Segmentation

Most segmentation algorithms which exploit textual information only are based on measuring similarity between potential segments (determined, e.g., by the cosine distance). Optimal segments have high intra-similarity (coherence) and low inter-similarity (differ from other segments) [17].

3.1.1 Similarity-based Algorithms

The two most often used algorithms for semantic segmentation are TextTiling [9] and C99 [4]. Both measure the similarity by calculating cosine distance between neighbouring segments. C99 calculates similarity between all sentence pairs using the cosine measure to create a similarity matrix and identifies regions with high intra-similarities along the diagonal of the matrix. TextTiling calculates the similarity for adjacent segments of predefined size and the points with the lowest values are designated as boundaries. This algorithm has been employed in Passage Retrieval before (e.g., [7]).

3.1.2 Lexical-chain-based Algorithms

Both similarity-based and lexical-chain-based algorithms make use of lexical cohesion in topical segments. The lexical-chain-based algorithms detect lexically related words – the amount of related words within one segment is typically higher than amount of related words between adjacent paragraphs. Lexical chain could be defined as "a sequence of lexicographically related word occurrences" [14, 27]. A segment boundary usually occurs at the point where large numbers of lexical chains begin and end.

3.1.3 Feature-based Algorithms

Feature-based algorithms make use of machine learning techniques which are applied to various features mined from data. An example of a common feature is a *cue phrase*. Ballantine [1] defines *cue phrases* as words and phrases which "serve primarily to indicate document structure or flow, rather than to impart semantic information about the current topic" (e.g., "Good evening", "well", "so"). Thus, they easily indicate the beginning or end of a segment. Beeferman

et al. [2] study effectiveness of various lexical features and show that the best feature is information on whether a given word was also present up to five words in the past. Other well-performing features include, for instance, presence of pronouns and named entities. The most effective features are then used to predict the probability that a topic ends at a given word or sentence and the decision on segment breaks is based on these predictions.

3.2 Segmentation in Audio-Visual Recordings

Compared to the text-based segmentation approaches, most algorithms for audio-visual recording segmentation are feature-based: they employ supervised machine learning techniques applied to various textual, acoustic, and visual features.

Hsueh and Moore [11] examine a range of features in a Maximum Entropy classifier and conclude that lexical features (i.e., *cue words*) are the most effective ones but they need to be combined with audio and visual features to achieve an optimal performance. As reported, other well-performing features include conversational features (such as silence, change of speaker activity, and amount of overlapping speech), followed by contextual features (dialogue act type and speaker role), prosodic features (e.g., fundamental frequency and energy level in audio track), and motion features (detected movements, frontal shots, hand movements).

Textual features in audio-visual segmentation need to be acquired using an ASR system. However, the quality of the transcripts usually varies and raises the question of how the quality of the transcripts influences IR. Hsueh and Moore [11] show that, despite the word recognition error of 39%, none of their systems performs significantly worse on ASR transcripts than on reference transcripts. They offer a possible explanation that the same word is misrecognized the same way in different parts of the corpus and thus, the cohesion is not influenced. Utilization of multimodal features could also reduce the impact of the transcript quality. The segmentation quality could also be improved by using lattices instead of a single one-best hypothesis of the ASR system: Mohri et al. [20] show relative improvement of up to 2.3%.

4. EXPERIMENTAL SETTING

All of our experiments were performed within MediaEval Benchmark;[1] an activity intended for development, comparison, and improvement of strategies for processing and retrieving multimedia content. We apply our methods in two tasks: Similar Segments in Social Speech (SSSS) Task and Search and Hyperlinking (SH) Task, organized in 2013.

The main aim of the SSSS task is to find segments similar to the given ones (query segments) in the collection of audio-visual recordings containing English dialogues of a university students' community. In the intended scenario, a new member (e.g., a new student) joins a community or organization (e.g., a university), which owns an archive of recorded conversations among its members. A new member wants to find information according to his or her interest in the archive to better understand the organization. The student wants to find more segments similar to the ones he or she is interested in and browses the archive using hyperlinks in videos to find more segments similar to the segment that he or she marks.

[1] http://www.multimediaeval.org/

	SSSS Train	SSSS Test	SH
Number of documents	20	6	2323
Hours of video	4	1	1697

Table 1: Statistics of the data collections.

In the scenario of the SH task, a user wants to find a piece of information relevant to a given query and then navigate through a large archive using hyperlinks to the retrieved segments. The main goal of the known-item Search Subtask is to find passages relevant to a user's interest given by a textual query in a large set of audio-visual recordings. Subsequently, in the Hyperlinking Subtask, the goal is to find more passages similar to the retrieved ones. In this paper, we focus on the Search Subtask only.

4.1 Test Collections

In this section, we describe the two test collections used in our experiments. Their basic statistics are presented in Table 1.

4.1.1 Similar Segments in Social Speech Data

The training and test data in the SSSS task consist of on purpose recorded interviews of two speakers. Several interview topics such as "movies" and "university studies" were suggested but the topics were not restricted. In addition to the manual transcripts created by the task organizers and automatic transcripts provided by the University of Edinburgh, the collection also contains detailed prosodic features and metadata (e.g., age, native language, and gender of the speaker, recording conditions for each document). The ASR transcripts are given for two tracks - one for each speaker. Therefore, we first merged these tracks into a single one, based on the timing.

The data consists of manually indicated segments which were also manually grouped into similarity sets. The collection is divided into training set and test set. The details of the task and data are described in [30].

We convert the task of searching all segments in the similarity set into a task of retrieving all segments similar to one given query segment. Each query segment is specified by the timestamp of its beginning and end. The actual queries are then constructed by including all words lying within the boundaries of the query segments. For each similarity set and for each segment in the similarity set, we consider each segment in the set as a query and the rest of the segments in the similarity set as the possible ground-truth points. Thus, the total number of queries is equal to the number of all segments in all similarity sets.

4.1.2 Search and Hyperlinking Data

The Search and Hyperlinking Task data collection consists of TV programme recordings provided by BBC.[2] The video recordings, audio track, metadata, synopsis, cast, detected shots, detected faces, visual concepts, subtitles, and two automatic transcripts, provided by LIMSI [15] and LIUM [23], are available together with the recordings themselves.

The subtitles given in the SH task also contain additional information and may thus also contain unuttered words, such as "SCHOOL BELL RINGS", "SNAP", and song lyrics: "# BOTH: It's wo-o-o-o-onderland... #". The timing in the

[2] http://www.bbc.co.uk/

transcripts is given on various levels, e.g., for the LIMSI and LIUM transcript, the exact timing of each word is given, for the subtitles and manual transcripts, the timing is specified for the whole utterances, which can contain several sentences. In these cases, we estimated the approximated word timing from the timing of the entire utterance and known number of words in the utterance assuming their equal duration.

For the Search Subtask, 4 training and 50 test queries are available. The queries consist of a *query text*, which is a short textual description of a relevant passage (e.g., "Boris Johnson"), and *visual cues*, which describe visual information of the query segment (e.g., "2 men sitting opposite each other").

The queries and relevant segments were defined by 29 users. Users marked relevant segments of any length and then formulated the queries. This process differs from the usual query input in which the user first specifies the query and then judges the retrieved passages. The reversed procedure could cause higher overlap of the queries and relevant passages because the users tend to use the vocabulary from the recording. On the other hand, the queries may be more diverse.

As the training set only consists of four queries, we collected another 30 queries by ourselves and used the whole set consisting of 34 queries for training. We randomly selected 29 recordings from the collection, identified short passages somehow interesting to us and formulated the queries to search for those passages (the passages of interest were then considered as ground-truth). The queries were formulated to imitate the style of the original given queries (e.g., "how to prepare Vietnamese spring rolls", "Thomas Tallis signature", and "a difference between a hare and a rabbit"). More details of the task and the data collection can be found in [6].

4.2 Evaluation Methods

We use Mean Reciprocal Rank (*MRR*) and Mean Generalized Average Precision (*mGAP*) to automatically evaluate our experiments.

MRR [29] is calculated as the average of Reciprocal Ranks over the set of queries. Reciprocal Rank for a single query is defined as the reciprocal value of the rank of the first correctly retrieved document. We apply *MRR* with two different settings: in the standard way on full recordings (denoted as *MRR*) and with starting points of retrieved segments limited to appear less than 60 seconds from the starting point of the relevant segment to be considered as correctly retrieved (denoted as *MRRw*). The standard *MRR* evaluates the quality of the retrieval of full recordings, whereas *MRRw* also takes into account the quality of the retrieval of the exact jump-in point.

mGAP is calculated as the average of *GAP* [16] values over the set of the queries. Similar to *wMRR*, the *GAP* measure also takes into account the exact jump-in point, which represents the beginning of a relevant segment. The quality of the jump-in point is assessed according to its distance from the beginning of the relevant segment using a penalty function. *GAP* for a single query is calculated as follows:

$$GAP = \frac{\sum_{penalty_k \neq 0} p_k}{N}, \quad (1)$$

where p_k is a precision at k and N is a number of ground truth points ($N = 1$ in the SH task). Each ground truth point can be used only once. $Penalty_k$ assesses the quality of the jump-in point and, at the position k, it is calculated as follows:

$$penalty_k = \frac{\sum_{i=0}^{k} penalty_i}{k} \quad (2)$$

where $penalty_i$ is estimated according to the penalty function, based on the distance between the starting point of the relevant segment and the starting point of the retrieved segment. The shape of the penalty function is triangular and depends on a given window width. We use the same window width of one minute in both *MMRw* and *mGAP*.

Seg	O	Manual transcripts			ASR transcripts		
		MRR	MRRw	mGAP	MRR	MRRw	mGAP
No	R	0.565	0.122	0.012	0.565	0.144	0.012
No	P	0.879	0.315	0.029	0.858	0.333	0.027
All	R	0.897	0.671	0.277	0.885	0.669	0.247

Table 2: Baseline results for the SSSS task. In column O, P refers to overlapping segments preserved and R refers to overlapping segments removed.

5. EXPERIMENTS

In this section, we compare several methods for segmentation of audio-visual recordings applied to the data for the MediaEval 2013 SSSS and SH tasks, described in Section 4.1

5.1 System Details

In all experiments, we employ the Terrier IR toolkit[3]. Based on the findings from our previous experiments [8], our system employs the Hiemstra language model [10] with the parameter expressing the importance of a query term in a document set to 0.35, stopwords removal, and stemming implemented in the Terrier system. The ranked lists of retrieved segments are post-filtered – the segments partially overlapping with either the query segment or a higher ranked segment are removed. In the SH task, both *query text* and *visual cues* are used to construct the queries. For both tasks, the system is applied to the ASR transcripts (LIMSI and LIUM for the SH task) as well as the manual transcripts (subtitles for the SH task).

5.2 Baseline Settings

Our main baseline runs are performed with no segmentation, i.e., each recording contains one segment spanning the entire length of the recording. The baseline results for the SSSS task are given in Table 2 and for the SH task, they are included in Table 4 (Row 1) together with other results. For the SSSS task, the main baseline (Table 2, Row 1) is compared with other results: Row 2 refers to the experiments without post-filtering (overlapping segments are preserved). This strategy outperforms the baseline experiment with post-filtering as in this case the recordings overlapping with the query segments are completely removed from the retrieval results. Row 3 refers to the experiments where the IR system is applied to the manually predefined segments. This can be viewed as a gold-standard segmentation and the scores as a theoretical maximum we could achieve with our IR system if the segmentation was completely correct. As

[3] `http://terrier.org/`

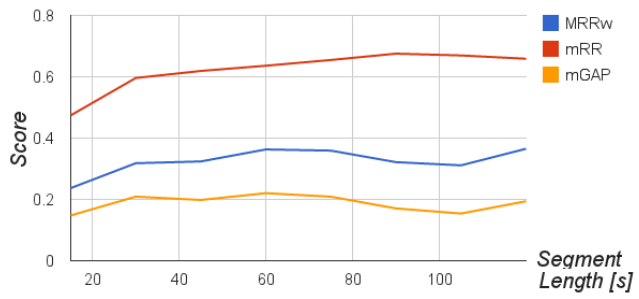

Figure 1: Evaluation scores vs. the length of segments applied in regular segmentation on subtitles in the SH task. The shift set to 15 sec.

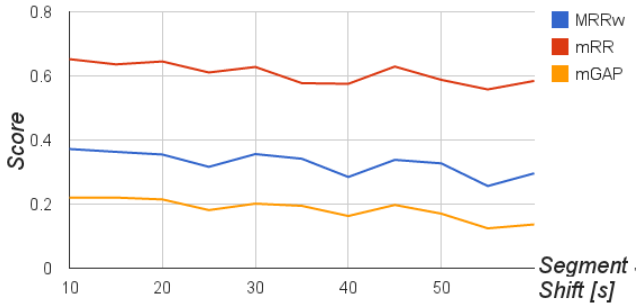

Figure 2: Evaluation scores vs. the length of segment shift in regular segmentation on subtitles in the SH task. The segment length set to 60 sec.

expected, the largest room for improvement can be seen in *MRRw* and *mGAP*, which take into account the exact starting points of relevant segments (cf. Table 2, Rows 1 and 3): The slight increase of the *MRR* score (from 0.879 to 0.897 for the manual transcripts and from 0.858 to 0.885 for the ASR transcripts) shows that applying the segmentation can improve retrieval of full recordings too.

5.3 Segmentation Strategies

In this section, we analyse the effect of segmentation on retrieval quality. We explore window-based segmentation and feature-based segmentation using a machine-learning (ML) approach based on decision trees.

5.3.1 *Window-based Segmentation*

In this approach, we experiment with sliding windows of different durations and overlaps. In comparison with the former approaches based on counting the number of words within the window [31, 13, 3], our strategy measures its time duration. The window of a particular duration is slid through the transcripts – it generates a new segment of the given length and is shifted by a particular time distance. The effect of varying the duration of the window (i.e., segment length) is drawn in Figure 1 and the effect of changing the window shift (i.e., segment overlap) in Figure 2.

In Figure 1, the shapes of the *MRRw* and *mGAP* curves are similar but slightly differ from the shape of the *MRR* curve. The highest *MRRw* and *mGAP* scores are achieved for segments 60 seconds long and the best *MRR* score is obtained using segments about 100 seconds long. Figure 2 shows that increasing segment shift consistently degrades the results (measured by all the measures) and the optimal segment shift is about 10 seconds. We did not experiment with segments shorter than 10 segments. In this case, the number of created segments would be too large and not feasible for large data collections.

5.3.2 *Feature-based Segmentation*

In this approach, we identify possible segment boundaries (beginnings and ends) using the J48 classification trees [21] implemented in the Weka framework[4] and trained on the training data available for the SSSS task containing manually marked segment boundaries.

We look at this problem as binary classification. For each word in the transcripts, we predict whether a segment boundary occurs after this word or not. In such a situation, distribution of the two classes (*segment boundary* vs. *segment continuation*) is highly unbalanced (more words appear inside segments than on their boundaries). Therefore, before training the model, we resampled the training data to change the class ratio (bias). We experiment with ratio values varying from 0.1 to 0.8, by taking as many instances of the *segment boundary* class as possible and as many instances of the *segment continuation* class as needed to achieve a particular ratio. The segmentation is trained on 66% of all examples randomly selected from the training data available for the SSSS task. The rest of the training set was used for segmentation tuning.

In the first experiment, we assume that each word in the transcripts belongs to a single segment and thus, the segments do not overlap. We use two variants of this approach. We first identify possible segment beginnings and assume that the previous segment ends at this beginning (further denoted as Beginning = ML and End = "–"). Then, in a similar fashion, we identify possible segment ends and assume that the new segment begins right after the detected segment end (denoted as Beginning = "–" and End = ML).

In the second experiment, we apply the same process to detect segment beginnings but assume that each segment is 50 seconds long (denoted as Beginning = ML, End = B+50). The segment length was estimated as the average length of manually detected segments in the training data of the SSSS task. Similarly, we use the ML model to predict segment ends and set the beginnings automatically (denoted as Beginning = E+50, End = ML). In this setting, the segments can overlap.

In the third experiment, we first identify all possible segment beginnings and all possible segment ends. Then, for each possible beginning, we identify the segment end (from the set of identified possible segment ends) which lies closest to 50 seconds from the beginning (Beginning = ML and End = ML).

5.4 Segmentation Model

This section describes the segmentation model including features, parameters and their tuning.

[4] `http://www.cs.waikato.ac.nz/ml/weka`

Beg	End	Manual transcripts					ASR transcripts				
		MRR	MRRw	mGAP	#Seg	Len	MRR	MRRw	mGAP	#Seg	Len
–	–	0.565	0.122	0.012	6	719.3	0.565	0.144	0.012	6	680.0
Reg	Reg	**0.858**	0.655	0.233	146	48.4	**0.834**	**0.615**	0.202	166	48.3
ML	–	**0.845**	0.626	0.231	1067	7.1	0.785	0.538	0.197	1659	3.5
–	ML	**0.858**	0.613	0.164	82	64.2	**0.809**	0.526	0.131	60	90.7
ML	B+50	**0.859**	**0.690**	**0.255**	1107	47.8	**0.818**	**0.623**	0.217	1933	48.5
E+50	ML	**0.865**	**0.677**	**0.247**	690	47.9	**0.820**	**0.616**	**0.226**	964	48.1
ML	ML	**0.844**	0.630	0.216	1425	46.1	0.779	0.538	0.153	2429	68.1

Table 3: Comparison of regular segmentation and several types of feature-based segmentation applied in the SSSS task. Best results and insignificantly lower results (according to the Wilcoxon signed rank test [32] at the 0.05 level) for each transcript are highlighted.

5.4.1 *Feature Description*

Our J48 classification model exploits the following features: cue words and tags, letter cases, length of the silence before the word, division given in transcripts (e.g., speech segments defined in the LIMSI transcripts), and the output of the TextTiling algorithm [9]. All the features are binary, indicating whether they appear or not; the length of silence is measured as a difference between timestamps of two adjacent words and is quantized into 15 buckets representing durations of equal length ranging from 100 ms to 1500 ms (by 100 ms) and the corresponding (binary) features indicating whether the length is longer that the corresponding bucket's value or not. The TextTiling feature indicates whether a segment boundary is detected by this tool after the current word or not.

The cue words were identified independently for beginnings and ends by automatic analysis of training data. We extracted two sets of words: words which frequently appear at segment boundaries and those which are the most informative for segment boundary (the mutual information between these words and segment boundary is high). In addition, we manually defined another set of words which did not occur in the training data but are supposed to frequently appear at the boundary. We also use features for cue word and tag n-grams (unigrams, bigrams and trigrams) appearing at a segment boundary in the training data (tagging performed by the Featurama tagger [26]). For each type of the cue features (word n-grams, tag n-grams, frequent words, informative words, and defined words for either beginning or end), there is an additional feature indicating whether at least one feature of the type occurs. Punctuation was removed from all transcripts and cue words and the transcripts and cue words were converted to lower case.

Our analysis shows that the most informative features are the division defined in the transcripts, the length of silence (especially if it is longer than 300ms, 400ms, 500ms, and 600ms), the output of the TextTiling algorithm, and n-grams of words and tags (especially the features indicating that at least one item of a set of words or tags is present). For instance, for segment beginnings, the word n-grams "if", "I'm", "especially", "the", "are you", "you have", and the tag trigram "VBP PRP VBG" are highly informative. The letter case feature seems to be informative for segment beginnings. For segment ends, the words "good", "the", "interesting" and "lot" (the article "the" appears in the list of the ending n-grams even though it cannot stand at the end of the sentence, which is probably caused by our approximation of word timing) are highly informative.

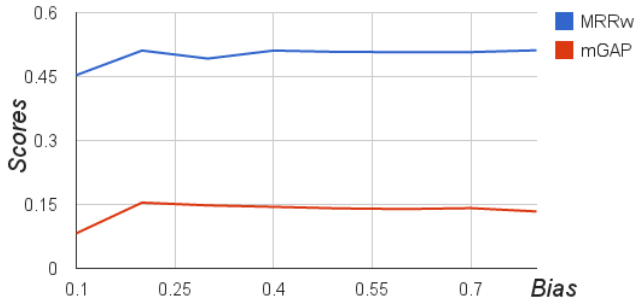

Figure 3: The *mGAP* and *MRRw* scores vs. class bias in the resampled traning data for the SSSS task, used for detection of ends of overlapping segments applied to the manual transcripts.

5.4.2 *Segmentation Tuning*

The segmentation model has several parameters which are tuned on the remaining 34% of the training data available for the SSSS task and kept aside for development purposes.

The J48 parameters are set as follows: the confidence factor is set to 0.25 by default for all experiments; the minimum number of instances per leaf is tuned on the development data for each experiment independently and varies from 2 to 250.

The final tuning parameter is the class bias in the resampled training data (*segment boundary* vs. *segment continuation*), which consequently affects the number of detected segment boundaries and also the retrieval quality. Figure 3 visualizes the effect of this parameter on the retrieval performance measured on the training data of the SSSS task by *mGAP* and *MRRw*.

Setting the Weka's parameter for class bias to 0.1 leads to around 7% of the development instances detected as segment boundaries with *mGAP* equal to 0.082. Increasing the bias to 0.2 improves *mGAP* to 0.154 (around 12% of the words were marked as boundaries) and then it slowly decreases. The *MRRw* score raises from 0.453 (for the bias equal to 0.1) to 0.511 (for the bias equal 0.2), then it stays almost constant, although there is a drop for the bias equal to 0.3.

5.5 Results and Discussion

We train and tune two J48 models – one to detect segment beginnings and one to detect segment ends. The models are trained on the manual transcripts of the training data

Beg	End	Subtitles					LIMSI ASR					LIUM ASR				
		MRR	MRRw	mGAP	#Seg	Len	MRR	MRRw	mGAP	#Seg	Len	MRR	MRRw	mGAP	#Seg	Len
–	–	**0.656**	0.052	0.027	2	2531.6	**0.553**	0.052	0.029	2	2589.6	**0.566**	0.050	0.028	2	2526.5
Reg	Reg	**0.671**	**0.388**	**0.245**	234	49.5	0.503	**0.299**	**0.172**	242	49.5	**0.535**	**0.275**	**0.169**	235	49.5
ML	–	0.549	0.117	0.060	3125	2.3	0.455	0.163	**0.119**	664	15.0	**0.556**	**0.246**	**0.134**	173	63.5
–	ML	0.607	**0.310**	0.192	280	29.0	**0.558**	0.180	0.102	20	293.5	**0.561**	0.121	0.051	9	854.9
ML	B+50	0.685	**0.412**	**0.272**	5820	49.6	**0.484**	**0.276**	**0.165**	748	49.6	0.424	**0.180**	0.095	171	49.3
E+50	ML	**0.715**	**0.428**	**0.298**	2580	49.6	**0.468**	**0.256**	**0.159**	435	49.5	**0.436**	**0.191**	0.087	74	48.3
ML	ML	0.626	**0.392**	**0.229**	5659	20.2	0.510	**0.250**	**0.141**	931	295.6	**0.501**	**0.201**	**0.123**	372	665.4

Table 4: Comparison of regular segmentation and several types of feature-based segmentation applied in the SH task for the ASR transcripts. Best results and insignificantly lower results (according to the Wilcoxon signed rank test [32] at the 0.05 level) for each transcript are highlighted.

applied to the ASR transcripts and the manual transcripts of the test data and compared.

As the training set used in the SH Search Subtask is very small (even after the additional training data is included), we decided to apply the SSSS-trained model in the SH task as well. This allows us to examine the possibility of creating a universal model for feature-based segmentation, however, at the same time, it also carries several potential problems. For instance: the sets of the cue words collected on the student dialogues may differ from the cue words used in TV programmes, different ASR systems may prefer different vocabulary, and finally, the silence between words in dialogues may have different distribution as the silence between words in the TV programmes.

The overall results for the SSSS task are displayed in Table 3. In the experiment with regular segmentation (Row 2), the recordings were divided into equilong segments of 50 seconds starting each 25 seconds (50% overlap). These values were set based on our experience from the MediaEval 2012 Search and Hyperlinking task [5] and the results discussed in Section 5.3.1.

In general, the best results are obtained by the feature-based segmentation into overlapping segments (Rows 5-6) which outperforms the regular segmentation applied to both types of transcripts measured by all three measures, except the *MRR* score on the ASR transcripts (but this measure is not sensitive to exact starting points of the retrieved segments). In terms of *MRRw* obtained on the manual transcripts, we even outperform the gold-standard segmentation (see Table 2). The results of the feature-based segmentation into non-overlapping segments (Rows 3-4) and the feature-based segmentation explicitly detecting beginnings and ends (Row 7) are consistently worse that those obtained by regular segmentation.

The results for the SH task, displayed in Table 4, are not as consistent as for the SSSS task and differ depending on the type of transcripts (subtitles, LIMSI ASR, and LIUM ASR). The feature-based approaches creating overlapping segments (Row 5-6) are effective especially when applied to the subtitles, where they outperform the regular segmentation in terms of all scores. However, for both ASR transcript, the regular segmentation is outperformed only in terms of *MRR*.

We should note that the segment counts and their average length differ substantially depending on the transcripts, even for the same segmentation approach. This is probably caused by the fact that the model was trained on data for a different task and the difference between the two types of transcripts it is applied to.

5.6 System Tuning

After the main experiments focused on segmentation strategies, we explored several options to improve performance of the best systems by other means. For the SH task, we tried to expand the segments by additional information provided in the collection. We concatenated the text from each segment with the metadata (title, source, variant, description, service name, episode name, and short episode synopsis), synopsis, and cast. Adding additional information improved the best result in the case when *visual cues* of the query were not used, otherwise the score dropped.

In the SSSS task, we explored several approaches to creating a query from the query segment and applied them on the regular-length segments. We constructed the queries both from manual transcripts and the ASR transcripts. The queries created from the manual transcripts achieved higher scores when applied to both the manual and ASR transcripts. We also tried to expand the queries by adding words appearing in the vicinity of the query segment (allowing from ± 5 up to ± 60 seconds) but none of these experiments led to better results.

6. CONCLUSION

In this paper, we have described our experiments with segmentation of recordings applied in two tasks of the MediaEval 2013 Benchmark: Search and Hyperlinking and Similar Segments in Social Speech.

We proposed and explored three types of feature-based segmentation employing decision trees to detect segment beginnings and ends. We described and analysed the set of features, the prediction model and its optimization. In general, our feature-based segmentation applied in the two tasks outperformed regular segmentation, which is claimed to be a very effective approach in the former experiments. The best approach in the SSSS task was the feature-based segmentation into overlapping segments of regular length. In terms of measures sensitive to exact segment starting points, the improvement was statistically significant on the manual (*MRRw* and *mGAP* measures) and ASR (*mGAP* measure) transcripts used in the SSSS task.

In the SH task, the results were not so conclusive, mostly because of the fact that the segmentation model was trained on data for the SSSS task which differ in many aspects. Still, some of the results are encouraging and confirm usefulness of our approach.

In future work, we would like to focus on improving segment coherence and thus reduce the number of segments and improve their content quality. We would also like to employ additional audio and visual features, such as video-shot segmentation (available in the SH task) and prosody information (available in the SSSS task), in the feature-based segmentation.

7. ACKNOWLEDGMENTS

This research has been supported by the project AMALACH (grant n. DF12P01OVV022 of the program NAKI of the Ministry of Culture of the Czech Republic), the Czech Science Foundation (grant n. P103/12/G084), and the Charles University Grant Agency (grant n. 920913).

8. REFERENCES

[1] J. Ballantine. Topic segmentation in spoken dialogue. Master's thesis, Macquarie University, 2004.

[2] D. Beeferman, A. Berger, and J. Lafferty. Statistical models for text segmentation. *Machine Learning*, 34(1-3):177–210, Feb. 1999.

[3] J. P. Callan. Passage-level evidence in document retrieval. In *Proc. of SIGIR*, pages 302–310, Dublin, Ireland, 1994.

[4] F. Y. Y. Choi. Advances in domain independent linear text segmentation. In *Proc. of NAACL*, pages 26–33, Seattle, WA, USA, 2000.

[5] M. Eskevich, G. J. F. Jones, R. Aly, R. Ordelman, S. Chen, D. Nadeem, C. Guinaudeau, G. Gravier, P. Sébillot, T. De Nies, P. Debevere, R. Van de Walle, P. Galuščáková, P. Pecina, and M. Larson. Multimedia information seeking through search and hyperlinking. In *Proc. of ICMR*, pages 287–294, Dallas, TX, USA, 2013.

[6] M. Eskevich, G. J. F. Jones, S. Chen, R. Aly, and R. Ordelman. The Search and Hyperlinking Task at MediaEval 2013. In *Proc. of MediaEval*, Barcelona, Spain, 2013.

[7] M. Eskevich, G. J. F. Jones, C. Wartena, M. Larson, R. Aly, T. Verschoor, and R. Ordelman. Comparing retrieval effectiveness of alternative content segmentation methods for internet video search. In *Proc. of CBMI*, Annecy, France, 2012.

[8] P. Galuščáková and P. Pecina. CUNI at MediaEval 2012 Search and Hyperlinking Task. In *Proc. of MediaEval*, Pisa, Italy, 2012.

[9] M. A. Hearst. TextTiling: Segmenting text into multi-paragraph subtopic passages. *Computational Linguistics*, 23(1):33–64, Mar. 1997.

[10] D. Hiemstra. *Using language models for information retrieval*. PhD thesis, University of Twente, Enschede, Netherlands, 2001.

[11] P.-Y. Hsueh and J. D. Moore. Combining multiple knowledge sources for dialogue segmentation in multimedia archives. In *Proc. of ACL*, pages 1016–1023, Prague, Czech Republic, 2007.

[12] M. Kaszkiel and J. Zobel. Passage retrieval revisited. In *Proc. of SIGIR*, pages 178–185, Philadelphia, PA, USA, 1997.

[13] M. Kaszkiel and J. Zobel. Effective ranking with arbitrary passages. *Journal of the Am. Society for IST*, 52(4):344–364, Jan. 2001.

[14] D. Kauchak and F. Chen. Feature-based segmentation of narrative documents. In *Proc. of ACL Workshop on Feature Engineering for ML in NLP*, pages 32–39, Ann Arbor, MI, USA, 2005.

[15] L. Lamel and J.-L. Gauvain. Speech processing for audio indexing. In *Proc. of GoTAL 2008, Advances in NLP*, pages 4–15, Gothenburg, Sweden, 2008.

[16] B. Liu and D. W. Oard. One-sided measures for evaluating ranked retrieval effectiveness with spontaneous sonversational speech. In *Proc. of SIGIR*, pages 673–674, Seattle, WA, USA, 2006.

[17] I. Malioutov and R. Barzilay. Minimum cut model for spoken lecture segmentation. In *Proc. of ACL*, pages 25–32, Sydney, Australia, 2006.

[18] C. D. Manning. Rethinking text segmentation models: An information extraction case study. Technical report, University of Sydney, 1998.

[19] E. Mittendorf and P. Schäuble. Document and passage retrieval based on hidden Markov models. In *Proc. of SIGIR*, pages 318–327, Dublin, Ireland, 1994.

[20] M. Mohri, P. Moreno, and E. Weinstein. Discriminative topic segmentation of text and speech. *Journal of ML Research - AISTATS Proceedings Track*, 9(1):533–540, 2010.

[21] R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, San Mateo, CA, 1993.

[22] G. Salton, J. Allan, and C. Buckley. Approaches to passage retrieval in full text information systems. In *Proc. of SIGIR*, pages 49–58, Pittsburgh, PA, USA, 1993.

[23] H. Schwenk, P. Lambert, L. Barrault, C. Servan, H. Afli, S. Abdul-Rauf, and K. Shah. LIUM's SMT machine translation systems for WMT 2011. In *Proc. of WMT*, pages 464–469, Edinburgh, UK, 2011.

[24] A. Smeaton, W. Kraaij, and P. Over. TRECVID - an overview. In *Proc. of TRECVID*, Gaithersburg, MD, USA, 2003.

[25] A. F. Smeaton, W. Kraaij, and P. Over. TRECVID 2004 - an overview. In *Proc. of TRECVID*, Gaithersburg, MD, USA, 2004.

[26] M. Spousta. Featurama – a library that implements various sequence-labeling algorithms. http://sourceforge.net/projects/featurama/.

[27] N. Stokes, J. Carthy, and A. F. Smeaton. SeLeCT: a lexical cohesion based news story segmentation system. *AI Commun.*, 17(1):3–12, 2004.

[28] J. Tiedemann and J. Mur. Simple is best: experiments with different document segmentation strategies for passage retrieval. In *Proc. IRQA Coling*, pages 17–25, Manchester, UK, 2008.

[29] E. Voorhees. The TREC-8 question answering track report, 1999.

[30] N. G. Ward, S. D. Werner, D. G. Novick, E. E. Shriberg, C. Oertel, L.-P. Morency, and T. Kawahara. The similar segments in social speech task. In *Proc. of MediaEval*, Barcelona, Spain, 2013.

[31] C. Wartena. Comparing segmentation strategies for efficient video passage retrieval. In *Proc. of CBMI*, Annecy, France, 2012.

[32] F. Wilcoxon. Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6):80–83, 1945.