Topic Segmentation of Educational Video Lectures Using Audio and Text

Markos Dimitsas[1] and Jochen L. Leidner[1,2][0000−0002−1219−4696]

[1] Coburg University of Applied Sciences and Arts, Friedrich-Streib-Straße 2, 96450 Coburg, Germany
[2] University of Sheffield, Department of Computer Science, Regents Court, 211 Portobello, Sheffield S1 4DP, UK

Abstract. The recent pandemic led to a surge of recorded lecture material available digitally, a resource that can now be used to improve computer-assisted learning. In this paper, we compare two methods for topic segmentation, i.e. the breaking down of a single lecture session into self-contained content units that deal with one or a small set of sub-topics or a set of concepts, respectively. We are interested whether auditory silence or keywords generated by a state-of-the-art keyword extraction tool are superior in segmenting down a session's recording into self- sufficient clips that may be served to student learners of artificial intelligence. To the best of our knowledge, this is the first comparison of silence-based topic segmentation and keyword-based topic segmentation for recorded lecture materials.

Keywords: Topic modeling · Topic segmentation · Detection of thematic shifts · Video analytics · Signal processing · Education applications.

1 Introduction

Recent increases in the acceptance of remote work, including remote lecturing, have led to substantial archives with lecture recordings that capture plenty of knowledge. In this paper, we describe an ongoing effort to design and implement methods for effective topic classification and segmentation of video lecture collections, in order to facilitate subsequent search (using a chatbot) and exploration (using a topic browser). While past work has established effective methods for text-based (e.g. [7]; [5]) and audio-based (e.g. c.f. [15]) methods, there is little work that combines modalities and exploits available thematic domain knowledge. Our work forms part of the *VoLL-KI* project ("Learning from Learners"), which aims to develop a toolbox of components that support learners of artificial intelligence and eventually other subjects [3], with a focus on the English and German languages. The remainder of this paper is structured as follows: Section 2 briefly summarizes past work in segmentation. Section 3 describes two methods, one using audio and another based on text; Section 4 presents our preliminary evaluation and related further plans for discussion. Section 5 discusses our findings before we conclude in Section 6.

2 Related Work

Topic segmentation (part of topic modeling) is a support task for navigating and understanding large documents or document collections. In traditional document topic segmentation, seminal works by Hearst[7] and Choi[5] laid the foundation for the field. Hearst's TextTiling algorithm is a pioneering method that automatically detects subtopics from expository text. It operates on the observation that a shift in topic is often accompanied by a change in the lexical distribution of a document. The algorithm consists of three steps: First, it tokenizes the text and creates sentence-sized units. Second, it determines a score for each of these units. Finally, in the third step, it detects subtopic boundaries. For the scoring process, three methods have been explored: blocks, vocabulary introductions, and chains. Each of these methods utilizes patterns of lexical co-occurrence and distribution within the text.

On the other hand, Choi's C99 algorithm takes a different approach to topic segmentation. Instead of focusing on lexical shifts, the C99 algorithm uses divisive clustering to detect boundaries in a document. Similar to TextTiling, it also consists of three steps. In the first step, pre-processing and sentence forming occur, along with the measurement of similarity between sentences, resulting in the creation of a similarity matrix. The second step involves ranking the similarity scores between sentences to estimate the order of similarity, thus creating a ranking matrix. The third step involves clustering to determine the location of topic boundaries. Initially, the entire document is considered as one coherent text segment, which is then iteratively divided to maximize the inside density.

More recently, approaches to topic segmentation[13, 2, 1] have integrated deep learning techniques: they utilize methods such as recurrent neural networks (RNNs), convolutional neural networks (CNNs) and transformers to capture sequential text dependencies and to model complex relationships, consequently enhancing both the accuracy and versatility of topic segmentation.

In the context of topic segmentation on lecture videos, there are multiple approaches to the problem. Most methods for segmenting lecture videos use textual information that is extracted from either audio (e.g., the textual transcript obtained by automatic speech recognition), visual (slide presentation), or a combination of both. Because of this, we can view the text-based part of the task as a problem of textual topic segmentation[6]. In [8] one of the first approaches to lecture video segmentation, they used a linguistic based approach since the existing algorithms for automated video segmentation relied on scene/shot change detection, something that lecture videos are lacking or have very few of. Also topic boundaries are less distinct due to the spontaneous nature of the lecturers speech. For that they propose an algorithm called *PowerSeg* that combines various linguistic segmentation features such as noun phrases, verbs, pronouns and cue phrases.

Shah et al. present TRACE [12], which is designed to perform automatic segmentation of lecture videos using a linguistic-based approach. It leverages Wikipedia articles and the lectures' video transcripts to create feature vectors from blocks of text. These blocks, created using a sliding-window architecture,

have a specific length and allow the system to skim through the entire documents. Afterwards, TRACE computes the similarities between the feature vectors of the Wikipedia article blocks and the transcript blocks. Transcript blocks that lead to the maximum similarity score which also exceed a similarity threshold δ are considered as a segment boundary.

In [6] they propose a lecture segmentation algorithm that extracts cue features from the lectures' video transcripts in an attempt to capture the essence of the text. Then these features are turned into vectors for representation purposes. Finally, a sliding window-based method is used to detect the segments in the video. The authors also introduced a new artificially-generated dataset for evaluation, consisting of synthetic lecture transcripts, as detailed in table 1.

In [10] the authors introduce VISC-L, a comprehensive framework that uses video transcripts for segmenting and characterizing videos, and linking them to their domain. Similar to previous methodologies, it employs knowledge models and a language model to identify primary topics and concepts for each video segment. Unique to VISC-L is its user study evaluation, which assesses the impact of the segmentation and characterization processes on concept learning.

However, relying solely on text for topic segmentation in lecture videos can overlook valuable information present in the audio and visual components of the video. This has led to the development of methods that incorporate audio features, such as silence detection and changes in speaker's tone, into the segmentation process. For instance, Malioutov et al. [9] proposed an unsupervised algorithm for topic segmentation that operates directly on raw acoustic information. Their method predicts topic changes by analyzing the distribution of recurring acoustic patters in the speech signal, demonstrating that audio-based segmentation can perform favorably even without input transcripts.

Moreover, some researchers have explored multimodal approaches, the combination of features extracted from video. These approaches aim to leverage the complementary information present in different modalities and enhance the segmentation process. For example, Soares and Barrere [14] proposed a multimodal approach that leverages both low and high-level audio features for automatic topic segmentation in video lectures. Their method combines frequency and power features from the audio signal, the transcript from automatic speech recognition and annotation features from a knowledge base. Through experiments on a dataset of Portuguese video lectures, they demonstrated that their method can successfully segment video lectures with various characteristics, and the results indicated that combining features from different modalities enhances topic segmentation performance.

Despite the extensive research in topic segmentation, our work introduces a unique perspective that has not been extensively explored in the existing literature. We examine both audio and text modalities individually for segmenting video lectures. While many methods utilize text, audio-visual cues, or their combination, we delve into the distinct strengths of audio and text in isolation. Notably, our use of keyword extraction for this task is a pioneering approach, contrasting the straightforward audio-based method with the intricate text-based

one, shedding light on the potential of different speech and text features for segmentation.

3 Methods

3.1 Silence-based Segmentation

The silence-based segmentation approach exploits natural pauses in speech that during a lecture may signify a transition from one subject to another. This methodology proves especially effective in educational video lectures, where typically a single speaker delivers the content, often accompanied by slide presentations. This format tends to encourage a structured pace and clear distinction between sections and topics. Furthermore, the audio quality in such settings is usually relatively free of noise, simplifying the task of identifying speech pauses. The process encompasses several steps, as outlined in Algorithm 1.

The first step involves extracting the audio track from the video lecture (line 2). We then identify pauses (lines 3-6) or silences in the speech by setting a threshold (line 4) and a minimum duration for silence (line 3), then automatically retrieving all the regions regions where the audio volume falls below this threshold (line 6) as pauses. This is done with the help of the pydub[11] library and its *detect_silence*() function. After identifying the pauses in the audio, we calculate their average length (line 7) and the Standard Deviation (line 8). Pauses that exceed the average length by more than σ standard deviations are considered significant and are marked as potential topic boundaries (line 11), thus creating the segmentations for the lecture video.

Algorithm 1 Audio-based Segmentation

1: **procedure** AUDIOSEGMENTATION(*video*)
2: $audio \leftarrow$ EXTRACTAUDIO(*video*)
3: $min\_silence \leftarrow 100ms$
4: $silence\_thresh \leftarrow dBFS - 16$
5: $silence\_params \leftarrow \{audio, min\_silence, silence\_thresh\}$
6: $silence\_list \leftarrow$ DETECTSILENCE(*silence_params*)
7: $silence\_mean \leftarrow$ CALCULATEMEAN(*silence_list*)
8: $silence\_std \leftarrow$ CALCULATESTDDEV(*silence_list*)
9: $selection\_criteria \leftarrow silence\_mean + \sigma * silence\_std$
10: $selection\_params \leftarrow \{silence\_list, selection\_criteria\}$
11: $silence\_selection \leftarrow$ SELECTSILENCES(*selection_params*)
12: **return** *silence_selection*
13: **end procedure**

3.2 Keyword Extraction-based Segmentation

Keyword extraction-based segmentation uses existing models or algorithms for extracting keywords from passages to segment text. For this implementation,

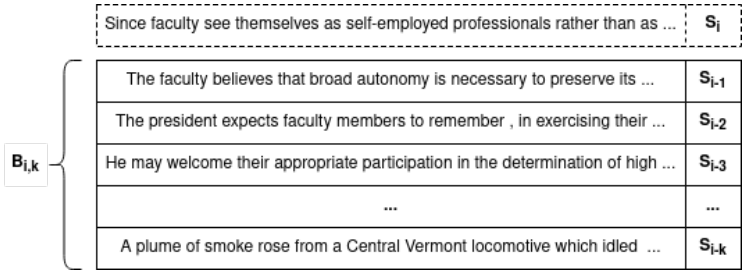

Fig. 1: This figure shows the keyword extraction-based approach applied on a document from the Choi dataset [5]

we used the YAKE! method [4], but theoretically this step can be reproduced by any existing keyword extraction methods, or even a vectorization method. The fundamental motivation behind this segmentation approach is put simply, is the hypothesis that a keyword for a given passage can be perceived as a summary describing the topic of that passage. Any change of the underlying topic (*thematic shift*) is expected to result in a change of the associated keyword describing of said passage. This concept can be used to test whether adjacent passages or blocks continue an existing topic or introduce a new, different topic. This is achieved by comparing whether their extracted keywords are equal or not, essentially comparing either individual keywords or sets of keywords.

The way that this idea is implemented is with a "sliding window" buffer architecture. The buffer B has a size limit of k sentences and each sentence i from the document gets compared with the buffer. At the beginning the buffer is empty, and after k iterations it is filled up. In the next iteration $(k+1)$, the first sentence is removed, and the $(k+1)$th sentence is added instead. Afterwards it continues in the same manner, similar to a FIFO queue, until all the sentences of the document will have been compared.

The buffer-sentence comparison is done for every sentence in the document, and it is done in the keyword level. For each iteration of the algorithm, a keyword extraction process is applied to both the buffer B and sentence S_i (line 6 & 7). So, for each iteration, the comparison is done between the keyword set of the buffer and the keyword set of the sentence. The number of keywords extracted from both the buffer and the sentence is kept the same to ensure a fair similarity calculation. For simplicity, we have chosen to extract a number of keywords equal to the buffer size, k. These sets may contain singular words as in keywords, or whole phrases, thus keyphrases. The choice between extracting individual words or phrases can be specified in the keyword extraction process using the *ngram* parameter. Apart from the parameter specifying the number of keywords to be extracted, we use the default values for all other parameters provided by YAKE! [4].In addition, YAKE! provides a relevance score for each extracted keyword, signifying its importance in the given text. This relevance score is used in the computation of the buffer-sentence similarity score.

Algorithm 2 Keyword Extraction-Based Segmentation

1: **Step 1: Calculate Gap Similarity Scores**
2: Set buffer size k
3: Initialize buffer B and empty list of gap scores $GapScores[]$
4: **for** each sentence S_i in the document D **do**
5: Extract k keywords $K[S_i]$ from S_i
6: Extract k keywords $K[B]$ from buffer B
7: Calculate sentence score using method A: $ScoreA(S_i, B)$
8: Calculate sentence score using method B: $ScoreB(S_i, B)$
9: Calculate sentence score using method C: $ScoreC(S_i, B)$
10: Take the maximum score: $ScoreMax(S_i, B) \leftarrow \max\{ScoreA, ScoreB, ScoreC\}$
11: Normalize sentence score $NormScore(S_i, B) \leftarrow 1 - ScoreMax(S_i, B)$
12: Add $NormScore(S_i, B)$ to list of gap scores $GapScores[]$
13: Add S_i to buffer B
14: **if** $length(B) > k$ **then**
15: Remove the first sentence from buffer B
16: **end if**
17: **end for**
18: **Step 2: Create Segmentation**
19: Set threshold θ
20: Initialize empty list of segmentations $Seg[]$
21: **for** each score $score$ in $GapScores$ **do**
22: **if** $score \leq \theta$ **then**
23: Append "1" to $Seg[]$ ▷ Mark as topic boundary
24: **else**
25: Append "0" to $Seg[]$ ▷ Mark as no boundary
26: **end if**
27: **end for**
28: **return** $Seg[]$

The similarity score for each iteration of the algorithm indicates if a sentence S_i has any similarity with the buffer $B_{i,k}$, i.e. the k previous sentences. If the similarity score is high, it indicates that the sentence is part of the same segment with the sentences of the buffer. If the similarity is low, then the sentence may be the first one of a new segment, indicating a topic shift. Because of the keyword extraction process used, we need to find a way to calculate the similarity between the two keyword sets.

The relevance score of each keyword contributes to calculate the similarity score for each sentence as follows: given the keyword set $K[B_{i,k}]$ of the buffer $B_{i,k}$ and $K[S_i]$ of the sentence S_i. There are three ways to calculate a similarity score (line 9-11), and the best one will be used (line 12).

1. If a keyword from $K[S_i]$ is found in $K[B_{i,k}]$ the relevance score of the keyword from the buffer set gets used for the similarity score. (line 9)
2. If for a keyphrase from $K[S_i]$, a word is found in $K[B_{i,k}]$ as a keyword, the relevance score of the keyword from the buffer set gets used for the similarity score. (line 10)

3. Lastly, for every word that exists in a keyword or keyphrase in $K[S_i]$ is compared with every word that exists in a keyword or keyphrase in $K[B_{i,k}]$. If similarities are found, then the relevance score is divided by the length of the keyphrase $partial\ relevance = kw\ relevance\ score/len(kw)$. (line 11)

After the score calculation, we obtain a list of similarity scores $Sim[S_1, S_N]$ for each of the sentences (line 13). Using a predefined similarity threshold T (line 5), we iterate through this list and begin placing segment breaks $Seg[S_i]$ (lines 14-16). If the similarity score $Sim(S_i)$ for a given sentence is lower than the threshold, this sentence is considered to belong to a new segment, and the algorithm places a segment barrier in front of it. Once all scores have been processed and all segment barriers have been placed, we end up with a segmented document. This is represented by a segmentation list $Seg[D]$ for the document D, containing the indices of the sentences that are preceded by segment barriers.

4 Towards an Evaluation

This chapter outlines the evaluation of the two topic segmentation methods for video lectures developed in this research. In the course of this research, we compiled a list of available datasets relevant to the task of topic segmentation (see Table 1). However, none were suitable for video lectures, thus posing a challenge in assessing our methods' effectiveness.

4.1 Evaluation Datasets

As can be seen from the table, some researchers [5, 6] synthesize evaluation data to overcome the lack of available datasets for their segmentation methods. While this is an ingenious approach to address the dataset scarcity issue, it can introduce biases and inaccuracies. Specifically, synthesized data often contain clear topic breaks due to the selection process and the way they are assembled. In contrast, real-world lectures typically feature more subtle topic shifts. Moreover, while synthesized data are usually in text form, which is relatively easy to create, synthesizing data for evaluating a video segmentation method using its audio is not straightforward and can be less accurate, making it less suitable for our purposes.

Given the lack of an existing dataset, we embarked on the evaluation process by manually annotating a video lecture. One of the authors, who also served as the lecturer, segmented the lecture based on slide changes, which served as a reliable indicator of topic transitions, in this case. His intimate understanding of the content guided the segmentation process, ensuring a high degree of accuracy in the annotated data. We acknowledge that evaluating on a single lecture (n=1) may not provide a comprehensive view of the effectiveness of our methods. However, given the constraints, we believe it offers valuable insights and serves as a starting point for further evaluations.

Table 1: Resources for Use in Topic Segmentation Evaluation – A Synopsis

Dataset	**Audio**	**Video**	**Text**	**Slides**	**Source**	**Lang.**	**Segmentation Annotation**
Hearst [7]	—	—	—	—	1 science article	en	—
Choi [5]	—	—	Orig	—	Brown corpus	en	Synthesized
Wikisection [1]	—	—	Orig	—	Wikipedia	en, de	Collected
ALV [6]	—	—	Auto	—	videolectures.net	en	Synthesized
SB18 [14]	✓	—	Auto	—	Brazilian lecture recordings	pt	Gold data
VoLL-KI [3]	✓	✓	Auto	—	recorded lectures	en, de	*Gold data (planned)*

4.2 Method Hyperparameters

Both methods in this study required hyperparameter tuning. For the silence detection method, the primary hyperparameter is the number of standard deviations σ from the mean silence duration, indicating topic boundaries. The keyword extraction method, however, requires tuning of the buffer size B and the similarity threshold θ. The buffer size influences the granularity of topic segmentation, while the similarity threshold determines sentence segment classification.

4.3 Grid Search

A grid search was employed to explore the hyperparameter space. For the silence detection, standard deviations ranged from 1 to 8. For keyword extraction, buffer sizes between 5 and 15 were tested, and similarity thresholds ranged from 0.05 to 0.95 in increments of 0.05.

4.4 Evaluation Metrics

Performance was assessed using standard metrics: accuracy, precision, recall, and the F1 score. These metrics quantify the effectiveness of each method in segmenting video lectures.

4.5 Evaluation Results

The grid search results and performance evaluations are visualized below. Included are two line graphs for the silence detection method (Figure 2) and four heatmaps for the keyword extraction method (Figures 3 and 4).

4.6 Analysis of Results

The results indicate that the optimal parameter for the silence detection method is a standard deviation value of 4, yielding an F1 score of over 40%. This performance significantly surpasses that of the keyword extraction method, which achieves a maximum F1 score of less than 10%. These findings suggest that

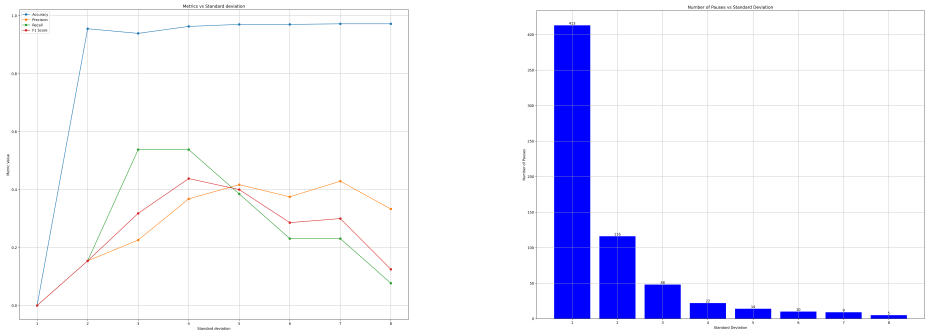

Fig. 2: Left: Performance of the silence-based segmentation method for different standard deviation values. Blue is Accuracy, Orange is Precision, Green is Recall, and Red is F Score. Right: Number of pauses selected as potential topic boundaries for different standard deviation values in the silence-based segmentation method.

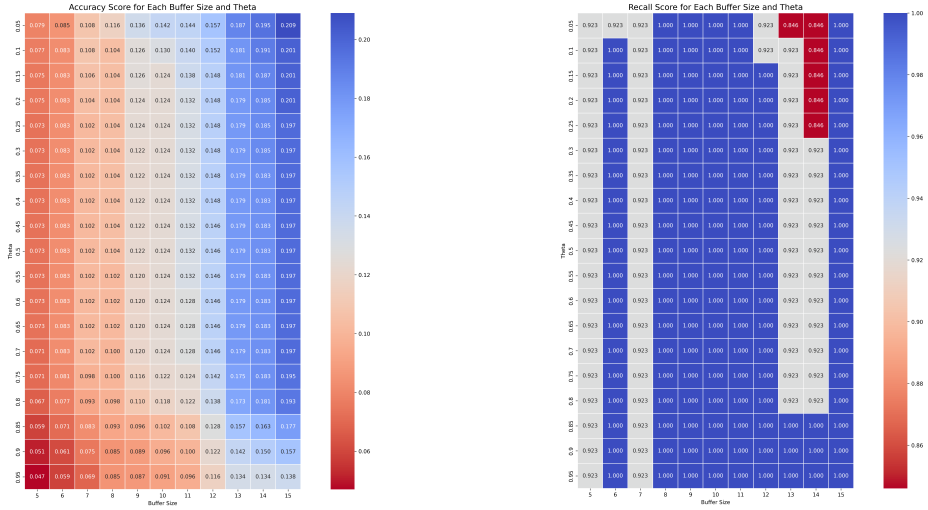

Fig. 3: Heatmaps showing the accuracy (left) and recall (right) of the keyword extraction-based segmentation method for different combinations of buffer size and similarity threshold.

the silence detection method, with its higher F1 score and inherent simplicity, might be more adept at segmenting video lectures into topics. Conversely, the keyword extraction method, despite its current lower performance, should not be dismissed outright. It's worth noting that this study does not demonstrate the ineffectiveness of keyword-based approaches as a whole. Rather, it underscores the need for additional work on this approach. The keyword extraction

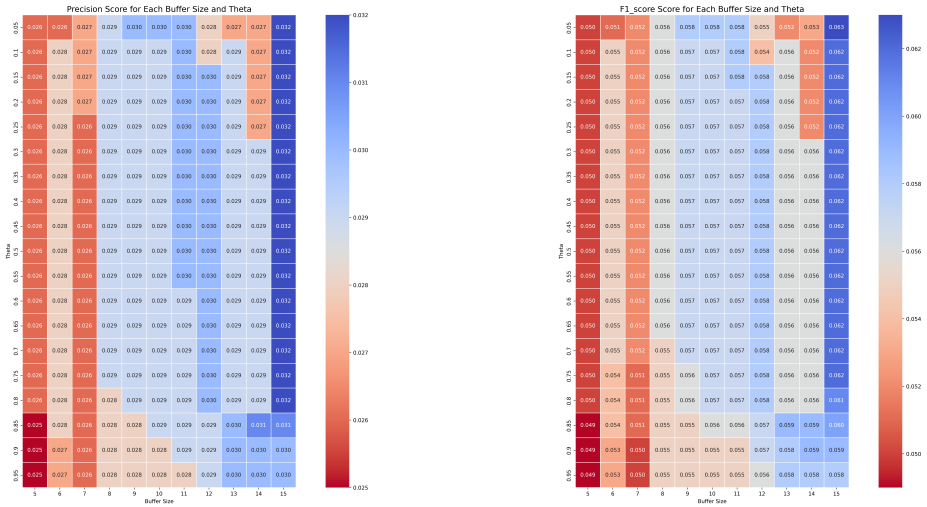

Fig. 4: Heatmaps showing the precision (left) and F1 score (right) of the keyword extraction-based segmentation method for different combinations of buffer size and similarity threshold.

method, with its complexity and intricacy, could be especially beneficial in scenarios where audio data is unavailable or when the audio quality is poor.

5 Discussion

The results of our study provide valuable insights into the application of silence detection and keyword extraction methods for topic segmentation in video lectures. The silence detection method, despite its simplicity, outperformed the more complex keyword extraction method. This suggests that the presence of significant pauses in speech, which can be easily detected and quantified, is a strong indicator of topic shifts in video lectures. In the context of our study, we found that the silence detection method, which has been used in conjunction with other methods in the literature, showed promise when used independently. This suggests that silence detection can be a useful tool for this task, even without additional methods. On the other hand, our novel application of keyword extraction for this purpose didn't perform as expected. One reason might be the challenge of comparing keyword sets for similarity. While keywords capture main topics, they might not be an effective medium to base the measurement of similarity between two sets, affecting the accuracy of segmentation.

Our study had several limitations. The most significant was the lack of an available dataset for evaluating our methods. We mitigated this by manually annotating a video lecture, but this approach has its own limitations, including potential bias and the difficulty of accurately identifying topic shifts. Furthermore, the use of a single lecture for evaluation limits the generalizability of our

findings. Despite these limitations, our study has important implications for the development of automated lecture segmentation tools. Our findings suggest that simple, easily quantifiable features of speech, such as pauses, can be effective indicators of topic shifts. This opens up new possibilities for the design of segmentation algorithms that are both effective and computationally efficient.

6 Summary, Conclusion and Future Work

In this study, we have presented two methods for segmenting video lectures into topics: a silence detection method and a keyword extraction method. Our evaluation results indicate that the silence detection method, with its simpler nature and higher performance, is more effective for this task. However, the keyword extraction method, despite its lower performance, may still have potential applications, particularly in cases where audio data is not available or the audio quality is poor.

Looking ahead, we plan to further enhance our topic segmentation methods. Moving forward, we plan to explore the use of vectorization methods, such as Word2Vec or BERT, as an alternative to the keyword extraction method currently used. Embeddings instead of keywords, could potentially capture more nuanced semantic relationships between words, thereby improving the accuracy of our topic segmentation. In addition, we plan to create a gold data dataset of annotated video lectures. This dataset will serve as a valuable resource for evaluating our methods and for benchmarking future topic segmentation methods. With this dataset, we will be able to retest our methods and potentially combine them into a new, more effective method for topic segmentation.

Acknowledgement

First, we are grateful to all lecture donors. The authors gratefully acknowledge the joint funding from German Federal Ministry of Research (BMBF) and the Free State of Bavaria for the Project grant "VoLL-KI: Von Lernenden Lernen" (Friedrich Alexander University (FAU) Erlangen, Coburg University of Applied Sciences/Otto-Friedrich-University Bamberg, under grants 16DHBKI089, 16DHBKI090 and 16DHBKI091) and to the funding to the second author by the Free State of Bavaria under the Hightech Agenda Bavaria R&D programme. All opinions are the authors' and do not reflect positions of the funding agencies.

References

1. Arnold, S., Schneider, R., Cudré-Mauroux, P., Gers, F.A., Löser, A.: SECTOR: A Neural Model for Coherent Topic Segmentation and Classification. Transactions of the Association for Computational Linguistics **7**, 169–184 (2019). https://doi.org/10.1162/tacl_a_00261
2. Badjatiya, P., Kurisinkel, L.J., Gupta, M., Varma, V.: Attention-based neural text segmentation. CoRR (abs/1808.09935) (2018), http://arxiv.org/abs/1808.09935

3. Berges, M., Kohlhase, M., Grubert, J.L.L.J., Landes, D., Mittag, F., Henrich, A., Nicklas, D., Schmid, U., vom Ende, A.U., Wolter, D.: VoLL-KI: Von lernenden lernen. Künstliche Intelligenz (2023, submitted), currently under review
4. Campos, R., Mangaravite, V., Pasquali, A., Jorge, A., Nunes, C., Jatowt, A.: YAKE! keyword extraction from single documents using multiple local features. Information Sciences **509**, 257–289 (2020). https://doi.org/10.1016/j.ins.2019.09.013
5. Choi, F.Y.Y.: Advances in domain independent linear text segmentation. In: Proceedings of the 1st North American Chapter of the Association for Computational Linguistics Conference. pp. 26–33. NAACL 2000, Association for Computational Linguistics, New York, NY, USA (2000)
6. Galanopoulos, D., Mezaris, V.: Temporal lecture video fragmentation using word embeddings. In: Kompatsiaris, I., Huet, B., Mezaris, V., Gurrin, C., Cheng, W.H., Vrochidis, S. (eds.) MultiMedia Modeling. pp. 254–265. Springer International, Cham, Switzerland (2019)
7. Hearst, M.A.: TextTiling: Segmenting text into multi-paragraph subtopic passages. Computational Linguistics **23**(1), 33—-64 (1997)
8. Lin, M., Chau, M., Cao, J., Jr., J.F.N.: Automated Video Segmentation for Lecture Videos: A Linguistics-Based Approach. International Journal of Technology and Human Interaction (IJTHI) **1**(2), 27–45 (2005), https://ideas.repec.org/a/igg/jthi00/v1y2005i2p27-45.html
9. Malioutov, I., Park, A., Barzilay, R., Glass, J.R.: Making sense of sound: Unsupervised topic segmentation over acoustic input. In: Carroll, J., van den Bosch, A., Zaenen, A. (eds.) ACL 2007, Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics, June 23-30, 2007, Prague, Czech Republic. The Association for Computational Linguistics (2007), https://aclanthology.org/P07-1064/
10. Mohammed, A., Dimitrova, V.: Video segmentation and characterisation to support learning. In: Educating for a New Future: Making Sense of Technology-Enhanced Learning Adoption: 17th European Conference on Technology Enhanced Learning, EC-TEL 2022, Toulouse, France, September 12–16, 2022, Proceedings. p. 229–242. Springer-Verlag, Berlin, Heidelberg (2022). https://doi.org/10.1007/978-3-031-16290-9_17
11. Robert, J., Webbie, M., et al.: Pydub (2018), http://pydub.com/
12. Shah, R.R., Yu, Y., Shaikh, A.D., Zimmermann, R.: TRACE: linguistic-based approach for automatic lecture video segmentation leveraging wikipedia texts. In: 2015 IEEE International Symposium on Multimedia, ISM 2015, Miami, FL, USA, December 14-16, 2015. pp. 217–220. IEEE Computer Society (2015). https://doi.org/10.1109/ISM.2015.18
13. Sheikh, I., Fohr, D., Illina, I.: Topic segmentation in ASR transcripts using bidirectional rnns for change detection. In: IEEE Automatic Speech Recognition and Understanding Workshop. ASRU 2017, Okinawa, Japan (2017), https://hal.science/hal-01599682
14. Soares, E.R., Barrére, E.: Automatic topic segmentation for video lectures using low and high-level audio features. In: Proceedings of the 24th Brazilian Symposium on Multimedia and the Web. pp. 189—-196. WebMedia '18, Association for Computing Machinery, New York, NY, USA (2018). https://doi.org/10.1145/3243082.3243096, https://doi.org/10.1145/3243082.3243096
15. Theodorou, T., Mporas, I., Fakotakis, N.: An overview of automatic audio segmentation. I.J. Information Technology and Computer Science **11**, 1–9 (2014). https://doi.org/10.5815/ijitcs.2014.11.01