BERTopic for Topic Segmentation of Multi-Party Conversational Data

Anja Škrlj

4th Year Project Report
Artificial Intelligence
School of Informatics
University of Edinburgh

2023

Abstract

This thesis investigates the value that topic modelling can bring to the task of topic segmentation for conversational data. I extend a topic segmentation algorithm (TextTiling) with a state-of-the-art topic model, BERTopic, and apply it to a conversational corpus. BERTopic is based on a language model trained on written, not conversational, data, which poses a challenge for this research. Therefore, the thesis also addresses conceptual questions about fitting document-level topic models to conversational data. Additionally, the challenge of fitting the model to the downstream application of topic segmentation is considered.

The topic model used is based on a pre-trained language model - I show a 2.5% improvement in the Pk score, against only using the language model in the same segmentation algorithm, showing that topic modelling can be useful for topic segmentation.

Table of Contents

Chapter 1

Introduction

Topic segmentation (TS) of conversational text has become an area of growing academic and commercial interest. The practical applications are wide-ranging and include improving speech-to-text transcription technology, enhancing text readability and clarity, and providing topic suggestions for AI writing assistants. Additionally, topic segmentation can assist in retrieving information from long or complex texts. While topic segmentation often involves labeling segments with topics, it is a separate task from topic modeling (TM), which focuses on assigning topics to documents and modeling the sub-topic distribution within a text; the two tasks are closely related.

Conversational data presents a challenge in natural language processing (NLP), due to it's noisy and unstructured nature, and a much smaller availability of data compared to written text. The most competitive results for both TS and TM on conversational data have been achieved by using contextual word embeddings [35, 15], though discourse coherence [39] and some dialogue-specific annotations [22] have also been used to improve performance. Given the recent interest in conversational analytics, a number of annotated conversational corpora have been developed in recent years - the annotations include dialogue-specific concepts and sometimes topic labels and boundaries.

This thesis will leverage a state-of-the-art topic model (BERTopic [15]) based on BERT [10] for topic segmentation of multi-party conversational data, and investigate how that compares to approaches using contextual embeddings, as well as older lexical models.

Since BERTopic is based on BERT embeddings, the two may carry similar information for the purpose of topic segmentation - so why should BERTopic be useful? Topic segmentation algorithms that do not utilise topic modeling have a local understanding of the topical cues within a text. Although these algorithms can recognize local shifts in language, they lack a broader understanding of the topics present. This is especially problematic when dealing with conversational language, which is noisy and lacks explicit topic-revealing cues. To improve the model's ability to handle the noise that is typical in conversational language this thesis presents a novel approach to topic segmentation that will answer the first and main research question: Can we leverage large pre-trained topic models to improve topic segmentation algorithms? An additional strength of topic model informed segmentation, is the possibility to assign labels to the

identified segments - this is left for future work, but highlighted here as an important reason for undertaking this research.

This research will be performed on data for which BERTopic is not trained. Large pre-trained language models are generally trained on written text, as there is a much higher availability of this data, compared with conversational data - this also makes them more suited to labelling the topics of written corpora. Additionally, topic segmentation, unlike topic modelling, is a sentence level task - we therefore aim to extract topics spanning only several utterances, as opposed to an entire document. An exploration of approaches to fitting the topic model to short segments of conversational data is required to answer the second research question: (2) how can we best fit document-level topic models to perform well on the downstream task of topic segmentation for conversational corpora?

This approach is also a way of leveraging already segmented conversational data which has been developed over the years: while this data is used extensively for evaluating models, little is done in harnessing this data for improving topic segmentation models. Fine-tunable models enable us to use this data to improve segmentation despite its limited quantities. This points to another question this thesis will tackle: (3) How can we leverage existing data, annotated with topic boundaries, to improve topic segmentation algorithms? While it is in my interest to see whether this data can be useful, this might have implications for domain transfer: while BERT-based methods do not require annotated data, some of my methods do - so why is this worth exploring? Topic segmentation in the domain of meetings is one with real-world applications for segmentation. Furthermore, there are other meeting corpora with segmented topics, like ICSI: an extension of my work, which uses meeting transcripts from a larger number of corpora, might give good results on meetings in the real-world. While this is not explored in detail in this work, this idea motivates the research despite the obvious domain limitation.

My approach using only BERTopic-supplied information improves the Pk and WindowDiff score of classical TextTiling by 7% and 13% respectively. An approach combining topic modeling with BERT sentence embeddings improves the Pk score against only using a BERT-based approach by 2.5%. The results of my experiments show that modern topic models can be useful for informing topic segmentation frameworks and improving their performance, and give interesting insight into fitting BERTopic to the downstream task of topic segmentation. This thesis is also the first work to use a large pre-trained topic model for topically-coherent segmentation.

The task of topic segmentation on the AMI corpus is illustrated in Table 1.1. This table shows a short excerpt of a meeting, it's topic labels, and the topic change that we are interested in predicting here. This example also highlights two previously outlined challenges. Firstly, the absence of explicit topical cues can be observed. For instance, while the topic of "evaluation of prototype(s)" is hinted at through mentions of "evaluation" and "criteria", it is never made explicitly clear and must be inferred by the participants based on the known context. Secondly, the significant amount of noise present in conversational language can be seen in the table. There are numerous noise words, such as "um" and "uh", as well as unfinished sentences or words marked

as [disfmarker].

Speaker	**Caption**	**Topic Label**	**Topic Change**
A	It is beautiful , and it's everything that we discussed .	presentation of prototype(s)	0
D	Yeah, I think it's a beautiful [disfmarker]		0
C	Oh thank you . [vocalsound]		0
A	Alright [vocalsound] what's next in our agenda ?	evaluation of prototype(s)	1
A	Um we're gonna discuss the evaluation criteria , and that's with Courtney .		0
D	Okay , it's a PowerPoint presentation . I don't really know exactly what we should uh talk about . It's under evaluation .		0
A	Right .		0
D	Alright . Um so these are the criteria we're gonna ask , is it easy to use , is it fashionable uh [disfmarker]		0

Table 1.1: Example of a meeting transcript from the AMI Meeting Corpus (ID ES2010d), as a sequence labelling problem. The topic change column is what we are trying to predict.

This thesis is divided into three main chapters: 'Background', 'Methodology and 'Results & Discussion'. I conclude with hopes for future work.

Chapter 2

Background

2.1 Introduction

This chapter will present the background of the task tackled in my thesis. We will look in depth at the two tasks we are concerned with: topic segmentation (TS), and topic modelling (TM). This includes an in-depth presentation of BERTopic - this model is an essential component of my own work. We examine separately the work done so far in applying topic models to segmentation, since that is the stream of research my contribution fits into. This is followed by a discussion on the evaluation metrics for the two tasks (TS and TM), for which the generic metrics of precision and recall are inadequate. Finally, we delve deeper into the challenges and specifics of the tasks for conversational data - as we will see, the majority of work is tailored to written text.

2.2 Topic segmentation

Topic segmentation is the task of splitting a text into topic-coherent segments and, optionally, assigning them a topic label. Many papers focus only on detecting the segment boundaries, which is a somewhat less complex task. In essence, it can be thought of as a binary classification task: every gap between 2 sentence-like units is examined and either labeled with 1 (boundary) or 0 (not a boundary).

This type of information about the structure of a document can be useful for a number of downstream applications. These include information retrieval [17], text generation [25] and summarisation [38]. For conversational language, the most direct downstream application are various transcription systems for meetings, interviews, etc. Additionally, writing assistants and chat bots might utilise this.

In the past, the lack of annotated training data was a great limitation to this research. Without data, focusing only on detecting the boundaries was more attainable. In 1997 a key unsupervised algorithm for topic segmentation, TextTiling [16], was introduced. This algorithm will be the basis for the topic segmentation algorithm proposed in this thesis - we will therefore examine it here in detail.

The intuition of TextTiling is that fixed-size blocks of text on either side of a sentence gap are compared to observe the extent to which the vocabulary changes in that gap, and placing boundaries on the basis of this similarity. In reality, fixed-size ”pseudosentences” are used instead of actual sentences. TextTiling requires no training and achieved impressive performance, although it was originally only evaluated on written text (magazine articles).

The original paper proposes several different ways of measuring the vocabulary change between sentences. The method which showed the best performance is referred to in the original paper as blocks, where the dot product of word counts of the windows on each side of a sentence gap is the measure of the similarity between the two windows [16]. In this thesis, the term ”classical TextTiling” refers specifically to this similarity metric based on word count. Afterwards, average smoothing is applied to the obtained graph of similarity scores at gaps. The similarity scores at gaps are transformed into a depth values at each gap, based on the distance from the peaks on both sides of the valley to that valley - this step is added to avoid the results being skewed by local changes within coherent topic segments. The transformation from similarity scores to depth scores is show in Figure 2.1, for an excerpt from a meeting from AMI. The part of the TextTiling algorithm shown in this figure will not be modified in this thesis, only the steps prior to this.

TextTiling remains a common baseline in topic segmentation research. Additionally, due to its' relatively simple idea, it is easy to modify with new scoring systems - this is one of the reasons it is appropriate for the purpose of this research.

Another unsupervised stream of research in topic segmentation works with representing a document as a semantic relatedness graph: in GraphSeg, nodes represent sentences, with edges connecting semantically related sentences. Maximal cliques of the relatedness graph determine coherent segments, and therefore the boundary positions [14].

Further efforts tackled topic segmentation and local topic labelling using reinforcement learning [36]. The topic assigned to the segments were from a hand-crafted set of possible labels. This allowed for local topic labelling without having any data annotated with topic labels. The hand-crafted labels can be a limitation, as they impact domain transfer: this approach achieved state-of-the-art performance on in-distribution data, but did not perform very well on other domains. A strength of this approach was that it combined topic segmentation and modelling: giving information on where topic-coherent segments lie, as well as the topic that they discuss.

In an effort to move research into the field of supervised learning tasks, extensive data sets annotated with topic labels and boundaries have been developed by processing Wikipedia articles - the sections and their normalised headings were used as the topic boundaries and labels [23]. This data was leveraged for improving both boundary detection and segment labelling. The SECTOR paper learnt topic labels from data in the following way: from 8.5k distinct Wikipedia headings, synset was used to normalise the headings into a set of 26 possible topics. The limitation here is the need for extensive training data, which can be difficult to obtain for many domains [2]. This model generalised well to unseen in-domain data but showed poor domain transfer capabilities.

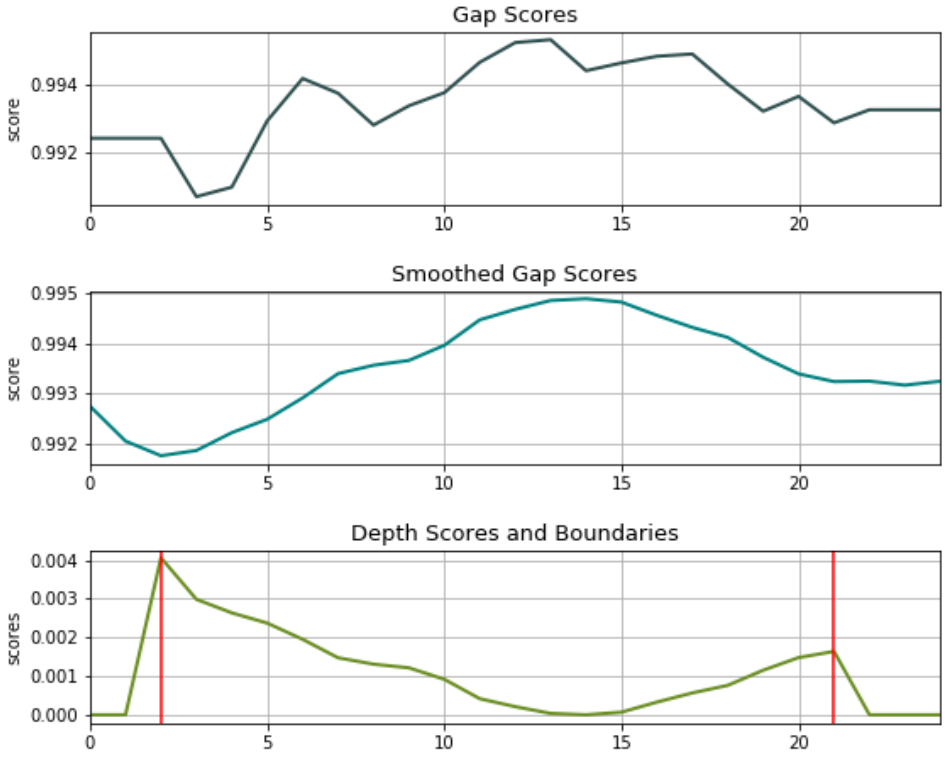


Figure 2.1: Graphs showing the steps of the TextTiling algorithm, after the similarity scores have been computed, up until the boundaries have been placed

Since the introduction of BERT (Bidirectional Encoder Representations from Transformers) [10], much of the research within natural language processing has been centered around leveraging this model for different applications within NLP. BERT is a pre-trained model for language representation that takes the context on both sides into account when creating the representation of a word. The model is pre-trained using unlabelled corpora with over 3.000M words - including the entirety of English Wikipedia. The data used in pre-training BERT consists exclusively of document-level written corpora. The model is relatively inexpensive to fine-tune and attained state-of-the-art performance on a number of downstream tasks [10].

Including BERT representations in topic segmentation algorithms has been shown to improve performance [22]. The TextTiling algorithm, due to it's simplicity and good performance, has seen enhancements using BERT representations of words and sentences: instead of measuring the similarity of segments with simple lexical co-occurence, metrics measuring the distance between the embeddings of the segments were proposed as carrying more meaningful information, successfully improving the performance of the model [35, 39, 9]. The work in this thesis is closely related to this stream of research, but we distinguish it by utilizing a BERT-based topic model instead of raw embeddings.

There are additional properties of language or their relationships, that have been shown to improve topic segmentation: an example of this is discourse coherence. Since topic segmentation is based on the relationships between 2 sentence-like units around the gap, modelling their relationship in terms of discourse coherence makes sense and has shown competitive performance [37, 39, 40]. This is to say that there are many properties of text we can look at to segment text by topic. We look to include the topic properties of text in this thesis and leave combining them with discourse to future work.

The metrics used for evaluating topic segmentation require annotated data to compare the model with. Precision and recall are rarely used in evaluation for this task: this is mostly because they penalise boundaries which are close to the reference segmentation. For example,a segmentation where each boundary is one sentence after a reference boundary accrues the same penalty as boundaries which are placed further from the reference boundaries at random. The common metrics used, which by-pass this issue are the P_k score and WindowDiff, elaborated on in more detail in the Evaluation section of this chapter.

2.3 Document Topic Modelling

Topic modelling is the task of assigning topics to documents and, in some cases, modelling the distribution of topics across a document. Notice that this is a different task to the segment labelling that is sometimes present in topic segmentation algorithms: there, the topics of shorter, topic-coherent segments are of interest; here, longer texts, that might span a number of sub-topics are considered.

Conventional probabilistic topic models, such as LDA [7], approached a document as a ”bag-of-words”: this is to say only word occurrence and frequencies were considered, whereas the information about the order and context in which they appeared were discarded. Given the bag-of-words, LDA assigns each word a topic, to model each document as a mixture of latent topics. This method requires the number of topics as a parameter, which requires good knowledge of the data and its' topics.

In practice, the context a word appears in carries important information. Effective contextual embedding techniques, like BERT, tackle this limitation of bag-of-words models, since BERT embeddings capture the nuances of a word's meaning indicated by the context. The same word appearing in two different contexts will give rise to two different BERT embeddings. [10] They have therefore been incorporated into, for example, LDA, showing improved performance. ([28, 31, 34]

Influenced by the recent interest and performance of word and document embeddings, a new stream of research has proposed that clustering word and document embeddings can produce meaninful topic representations. To motivate this, Angelov [1] proposes viewing topics as continuous instead of discrete values: as such topics are just points in the semantic space. If the embeddings we are using are designed to convey semantic similarity, he suggests we can compare embeddings of documents and words, thereby also topics and the distances between them, despite the fact that they represent different concepts. These clustering methods, as well as the assumption about semantic space

are the basis for the state-of-the-art model: BERTopic, which will be leveraged in this work.

BERTopic [15] uses BERT and TD-IDF to output a representation of the topics of the document. The intuition follows the clustering methods described above. Documents fed to BERTopic are embedded using Sentence-BERT. A density-based clustering algorithm identifies clusters of documents - each of the resulting clusters constitutes a topic. TF-IDF is used to extract representative words from the topic clusters.

It is a dynamic topic model, so it is developed to analyze the time evolution of topics in document collections [6]. A global representation, independent of time is generated first, and then local representations - it can also be clustered according to other criteria, not only time. It is notable that this model was evaluated on a big corpus of tweets (Trump's tweets), showing it performs well on informal text in short segments. This is especially relevant for potential applications to conversational corpora, and is a major reason why this model is appropriate for my research. A more detailed presentation of BERTopic will be given in section 2.4.

In the deep learning era, neural topic models are on the rise. An important idea researched within this stream is capturing topic correlations. This includes pair-wise relations in covariance matrices [27], and tree structures of topics[20]. Notice, that LDA does not capture this, and treats topics as independent. BERTopic fits into this category - the way in which relationships between topics are captured in BERTopic will become clear in later sections. However, these relationships will not be accounted for in my approach to topic segmentation, for reasons of simplicity.

2.4 BERTopic

There are several advantageous properties of BERTopic that make it the best choice for the purpose of my research. BERTopic has achieved state-of-the-art performance on conversational data, as well as on very short documents, like tweets; this makes it the most powerful topic model we could use here [15]. Moreover, The model can be fit to small amounts of data, which will allow us to leverage existing labelled data to improve segmentation performance.

In this section I describe the steps taken by BERTopic during fitting: by default BERTopic also trasnforms the documents used for fitting into their topic representations. Note that there are no pre-trained topic representations, so this fitting step is not optional.

Fitting BERTopic requires a list of *documents* - note, a document here need not mean an actual traditional document. A document here, refers to any type of textual input we want BERTopic to model - anything from an academic paper to a tweet.

In the first step of fitting, it converts each document to an embedding representation. It uses Sentence-BERT (S-BERT) for this by default, but can be made to use any embedding which is fine-tuned for capturing semantic similarity. Before the documents are clustered, UMAP dimensionality reduction is performed to overcome the issues of high-dimensional data for clustering.

The document embeddings are then clustered using Hierarchical DBSCAN: a density-based method for clustering. This method partitions the space into regions with similar density and clusters the data points (documents) accordingly. The topic that a specific document is labelled with corresponds to the cluster it was assigned to.

Documents which are closer to the average of the topics then any individual one, are assigned a topic of "-1": the documents belonging to this topic also form a cluster, but this topic is assumed to be uninformative.

Finally, a class-based variation of TF-IDF is used to extract representative words from the document cluster. This is done by concatenating all documents in a cluster into a single document, then computing the value:

$$W_{t,c} = tf_{t,c} * \log\left(1 + \frac{A}{tf_t}\right)$$

Where $tf_{t,c}$ denotes the frequency of term t in class c, A denotes the average number of words per class and tf_t the overall frequency of term t. The second term of this multiplication is the global IDF (inverse document frequency) value. The result $W_{t,c}$ is a measure of the importance of term t for the class c. The classes in this context correspond to the clusters formed in the previous steps of BERTopic. The terms with the largest $W_{t,c}$ value for some specific class c are the representative terms for this topic - a list of these words will form the human-readable representation of the topic. At this stage the set of possible topics is known, in additional to the topic assignments for each document.

The fit BERTopic model can now transform any new documents or segments: it compares the embedding of the inputs with all the topic clusters, and assigns it the topic that it is closest to. An important feature enabled, that will be useful for the experiments performed here, is that BERTopic can output the probabilities of each topic, instead of just a single topic ID. This is important because, as we will see later, BERTopic finds very granular topics in the corpus, and segmenting into topics that specific is not valuable. Instead, we want to think of the annotated topics as having specific distributions across the more granular topics. Given this intuition, supplying the probability distribution across the possible topics will carry information that is better suited to the task at hand. It is important to note that the representative words of a topic serve to represent the topic in a human-interpretable way, but the topic also has a representation in the embedding space. In assigning new documents a topic, the topic vector representation is compared with the document embedding, not the words that are used to represent it.

Additionally, notice that the model only creates topics and their representation during this fitting step, so there is no way to transform inputs without first fitting BERTopic on some data, thereby learning a set of topics. Once fit, the model can transform any input into topic probabilities, but will only generate meaningful outputs when the inputs are similar in content and form to those used during fitting. We have also seen that in fitting, the inputs are transformed into their desired form: this means we could fit BERTopic to the segments we are currently transforming, if no prior fitting on larger or labelled data was possible. The minimum number of documents, required for the fitting to work is

10, which is a low enough number to not limit us: however, from only 10 strings we would not expect the learnt topics to be useful.

2.5 Guided BERTopic

BERTopic has a setting allowing guided topic modelling, in which a list of topic representations, called a seed, is input by the user before fitting. The documentation suggest that the seed should contain a list of words for every known topic we want BERTopic to recognise. Each of the seed topic representations is embedded using S-BERT - the same model that is used to embed the fitting documents.

Each of the fitting documents is assigned to one of the seed topics, based on the cosine similarity between the seed and document embeddings. Each document embedding is averaged with the embedding of the seed it was assigned. In this way the seed informs the way the clusters are formed, encouraging them to be formed around the seeds. If a document is closer to the average of the seeds then to any of the individual seeds, it is assigned a label of -1 and is not averaged with any of the seed embeddings.

In the c-TF-IDF stage of BERTopic fitting, the IDF value of seed words is increased using a uniform multiplier, the default being 1.2. The IDF value can be understood as a penalty multiplier given to words for their frequency - increasing the IDF value of a word therefore boosts the word's importance, making it more likely to represent a topic.

Notice that it is unusual to compare the embeddings of different concepts in this way: it is a bold assumption to make that the seed topic embeddings and document embeddings will reside in the same space, given how different the form of the input is. Due to the possible inadequacy of this assumption, guided topic modelling with BERTopic may prove to be a flawed approach.

2.6 Topic Modelling for Topic Segmentation

Successful efforts have been made to leverage the information provided about a document by topic models to improve or create topic segmentation algorithms. The approaches to this broadly fit into one of two categories.

The first is the group of Bayesian approaches, which often perform both tasks of topic modelling and segmentation within the same algorithm. These approaches use a generative model, sampling topics and boundary indicators, based on some prior topic and topic shift distribution assumed in a document. Bayesian topic segmentation treats words in each segment as being generated from a segment-specific language model. Later extensions add a form of hierarchy: for example, Du et al. (2013) [12] approaches a document as having some distribution of topics. From this document-level distribution, a segment-level distribution of topics can be sampled. The individual words in the segment are then sampled from the segment-level distribution. This is an interesting intuition about the topics of documents, that will aid in the interpretation of our own method later on.

The second approach inserts existing topic models into topic segmentation frameworks to measure segment similarity. This has been done in several papers with LDA topic modelling[30], which is no longer a competitive topic model - it has been surpassed by neural and BERT-based approaches. Riedl et al. (2012) [33] used LDA to extend several topic segmentation algorithms, including TextTiling. The resulting algorithm, TopicTiling, measured the similarity between windows at sentence gap by comparing the topic ID assigned to them by LDA. They showed that topic models can improve performance of existing topic segmentation models. This method outperformed the state-of-the-art unsupervised segmentation algorithms at the time, and was faster then previous methods leveraging LDA. My work belongs to this stream of research, updating it by using a state-of-the-art topic model '

To my knowledge, there is no research using state-of-the-art topic models for topic-coherent segmentation.

2.7 Evaluation

The evaluation of a topic segmentation algorithms relies on two main metrics: Pk and WindowDiff. The development of Pk was motivated by the inadequacies of precision and recall in capturing the closeness between the hypothesized and reference boundaries of a document.

The metric operates by examining pairs of sentences at a distance k from each other to determine if they are correctly labeled as being in the same or different segments, as shown in 2.2. k is set to half of the average true segment size, and the process is repeated for the entire text. The returned result can be thought of as the probability that a randomly chosen pair of words which are k words apart is classified incorrectly.

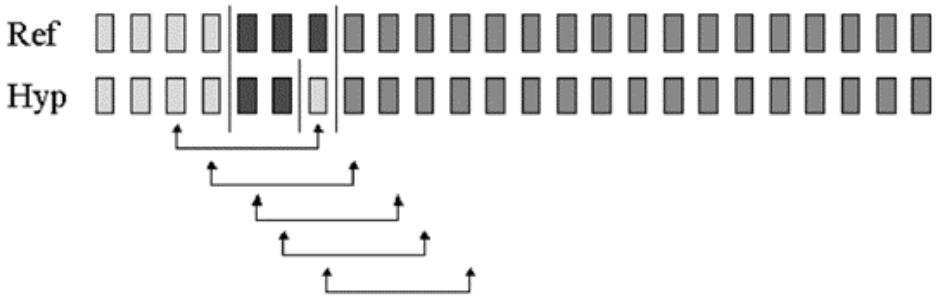


Figure 2.2: This figure illustrates the Pk metric on an example reference and hypothesis segmentation. The solid arrows in this image give rise to a penalty of 0. [32]

While Pk has proven useful, it has some limitations. For example, it is easy to see from figure 2.2, that a false positive that is close to an actual boundary, will accrue no penalty.

An improved metric of WindowDiff was proposed to tackle this shortcoming[32]. This metric compares the number of segment boundaries that fall in the interval between two sentences, rather than evaluating whether the sentences are in the same segment. When the number of boundaries in the window is unequal, penalty is given. In the

situation illustrated in figure 2.2, a penalty is given to the false positive boundary. This approach also tackles other issues of the P_k metric, identified in the paper proposing WindowDiff. This includes over-penalising false negatives and the sensitivity to segment size. WindowDiff also takes the hyper-parameter of k, which governs the size of the interval analysed, but has no conventions about the ideal value for this parameter.

In both metrics, the penalties are divided by the number of gaps to give a value between 0 and 1. We can think of this value as the probability of getting the ”relationship” (are they in the same or a different segment) between two random sentences wrong - a smaller value therefore implies better performance. Both of the mentioned metrics will be used in this work, for the purpose of consistence with previous research, which typically evaluates models using both metrics.

The evaluation of topic models will also be relevant in this paper. Since topic modelling is hard to evaluate on annotated text, as there is no single correct answer, alternative methods are used: topic coherence and topic diversity.

Topic coherence is an automated metric which is intended to approximate human 'interpretability'. It is oft evaluated using normalized pointwise mutual information (NPMI) [8], which has been shown to imitate human judgement with reasonable performance [24]. The NPMI is a value betwen -1 (incoherent topics) and 1 (perfect association).

Topic diversity compares the topics to one another and measures whether they are similar to one another. This penalises sets of topics where a number of different topics would contain the same word, indicating redundancy. Specifically, it is measured as the percentage of unique words for all topics, returning a value between 0 and 1, with 1 indicating more varied topics. [11].

In exploring the different approaches to fitting BERTopic, these two measures will be used for analysis. The evaluation measures show that these models are quite general, so while the topic assigned to a document might be ”correct” in some sense, it may not be what we are looking for in some specific application - for this reason, downstream application performance can often be a viable evaluation metric - in our case, the downstream task in question is topic segmentation.

2.8 Challenges for Conversational Language

Topic segmentation and modelling for conversational data has gained significant traction in recent years. Some dialogue-specific downstream applications for these tasks include functionality for AI writing assistants, and the improvement of the clarity and readability of automated text-to-speech transcription. Additionally, accurate segmentation of conversational text can be used for information retrieval from meeting or interview transcripts.

Any natural language processing task is faced with several challenges when confronted with conversational data. Difficulties include the use of informal language, a higher degree of fragmentation and a different set of coherence relations than written docu-

ments. Topics and arguments in conversations are less compact and do not follow a pre-ordained structure, which written text often does.

In addition, there is the limitation of lack of conversational data available, especially data that could be automatically segmented, the way WikiSECTION [2] segmented written corpora. This has been tackled by using a weakly-supervised approach to dialogue segmentation and modelling: prior knowledge is used to noisy label some data, which is used by the reinforcement learning algorithm - this works because RL can keep refining itself after the training with noisy data. [36]

Efforts have been made to fill this gap in conversational data, with a number of annotated conversational data sets being developed, mainly in English and Chinese. The development of these data sets can be expensive and labour-intensive [26, 41]. Since little naturally-occurring conversational data is available, some corpora include synthetic dialogues or ”acted” scenarios: for example, a part of the AMI meeting corpus was developed by giving participants roles and a task, then recording their meetings, as opposed to only recording naturally-occurring meetings in the real world [29]. Conversational corpora are split into several categories based on the number of speakers and intent of the conversation. This thesis is concerned with multi-party dialogue in professional settings (meetings): the AMI and ICSI [21] corpora are the biggest corpora of this kind in English. It is the AMI meeting corpus I will primarily use in this thesis. The corpus contains topic annotations which will be used to inform the models developed, as well as in evaluating their performance on topic segmentation.

The AMI corpus has been used to evaluate a number of topic segmentation algorithms - this offers ample points of comparison for our research. In fact, both classical TextTiling and enhanced versions of TextTiling have been applied to AMI. Since the approach proposed here is also an extension of TextTiling, this will enable direct comparison of of the topic-segmentation relevant information carried by the output of the topic model, compared to other sources of information.

Conversational data is often annotated with dialogue-specific labels, such as speaker, utterance intention [22], Dialog act (DailyDialog) - which are often important markers of topic change. Using these to improve the segmentation of conversational text has shown competitive performance. Zhang et al. (2019) approached the specifics of dialogue by treating it as a stream that develops in chronological order - a temporal convolutional network, which allows no look-ahead was used [42].

As with other natural language processing tasks, BERT has been leveraged for analysing the topics within conversational data, and showed improved performance. Since pre-trained signals (like BERT) are learned on written data, they do introduce some noise into conversational data [39] - despite this limitation, BERT remains the state-of-the-art model for many downstream applications, even for conversational corpora.

This limitation is also applicable to BERTopic, since it is based on a language-model trained on written corpora. However, BERTopic has achieved state-of-the-art performance on shorter length and noisy data, including a corpus of Trump’s tweets. BERTopic has been used for topic modelling in dialogue and shown best performance out of contemporary models [13][18], which motivates my use of it.

Chapter 3

Methodology

3.1 Datasets

The AMI meeting corpus is used both to fine-tune BERTopic and to evaluate its' performance for topic segmentation. The corpus is an appropriate one for this thesis because it is annotated with topic labels and boundaries. By virtue of the existing annotations, there is also existing topic segmentation work on AMI - including the application of TextTiling. This is useful as a point of comparison for our model's performance. Additionally, meeting transcripts offer the most in terms of real-world application of conversational topic segmentation, hence the focus on meetings in my work.

The AMI meeting corpus consists of 100 hours of meeting recordings and their transcripts. Roughly a third of the corpus consists of naturally occurring meetings; in the remaining two thirds, participants play different roles in the design team, completing a design project from start to finish in the course of one day. When partitioning data into the test, development, and training sets, the different types of meetings are distributed randomly amongst them.

AMI comes with a wide array of annotations and multi-media content. For this thesis, the topic annotations will be of the greatest importance. There are 24 distinct topic labels, which are split into 3 categories:

- functional topics (for example: closing, chitchat),
- top level topics (examples: new requirements, marketing expert presentation) and
- sub-topics (exampled: project budget, existing projects)

There is optionally also the category of "other", for topics in meetings that may not correspond to the categories they were given. Due to the different categorizations of the topics, they are often overlapping. For some of the approaches, this may be problematic and make the segmentation more difficult. The details of this in specific experiments (or baseline experiments) will be discussed and justified in the appropriate sections.

Before fitting BERTopic on the AMI corpus, noise-reducing pre-processing is required

Data Set	Length	Length After Pre-processing
Training	227980	117979
Test	192957	100431

Table 3.1: The length in words of the training and testing set, before and after the pre-processing step.

- the transcripts are completely faithful to the recordings, in that they also include all the filler words (for example, ”hmm”, ”umm”, etc.), and these words occur often. I manually composed a list of the occurring fillers, by inspection of the data, and removed them in the pre-processing step. Additionally, following the example of other uses of both TextTiling and BERT on the AMI corpus, I removed the stop words, as appearing in the *nltk* corpus (for example, ”the”, ”a”, ”that”). Utterances that only contained those words were therefore removed entirely.

Our model is evaluated against an AMI segmentation which observes boundaries from two of the topic categories: ”top-level” and ”subtopic”. This accounts for roughly two thirds of the boundaries present in the AMI corpus. The same pre-processing is done before testing the model on AMI meetings. Notice that this may influence the number of gaps in a segmentation and remove gaps at which reference boundaries lay - however, since removed utterances carry no meaning, we do not care if they are on the correct side of the boundary, so no important information is lost by doing this.

Table 3.1 shows the number of words in the training and testing set, both of which are (non-overlapping) subsets of AMI. This illustrates the size of the data sets we are working with, and highlights the amount of noise present in this conversational corpus - nearly half the words in the meetings are filler words or stop-words.

Note that this data has other annotations, which are potentially meaningful for the task of topic segmentation. This includes speaker roles, speaker shift, intent and gestures. Speaker shifts especially could be useful for a real-life application of segmenting automated transcripts of meetings, since a speech-to-text API could probably easily pick up on them. However, since the purpose here is not to build a state-of-the-art system, but rather to investigate the value of a large topic model, this data will be disregarded for the purpose of my work.

3.2 Evaluation

Due to the inadequacy of precision and recall, the standard evaluation metrics for topic segmentation are Pk and WindowDiff, as outlined in section 2.5. They both take a parameter k, which is the size of the sliding window used.

The value of k in Pk was set to half of an average segment length in the reference segmentation, as per the recommendation in the original paper [4]. The k is calculated separately for every meeting that the Pk was calculated for, by dividing the length of meeting with the number of topic boundaries - this is the standard implementation in

the *nltk* library.

WindowDiff has no guidelines on the ideal choice of window size - we set the value of k for this metric to 10 in our experiment.

3.3 Baselines

This section will describe the baseline experiments that my own work will be evaluated against.

The first two baselines are entirely naive approaches proposed by Beeferman et al. (1997) [3]: the *Even* baseline, in which a boundary is placed at every n-th sentence gap, with the value of n set to 30; and the *Random* baseline, which places boundaries uniformly at random across sentence gaps.

The other two baselines used are based on TextTiling, which makes them perfect for comparing the amounf of segmentation-relevant information carried by BERTopic. The first TextTiling baseline is classical TextTiling - a simple method evaluating vocabulary change between blocks [16], based on the dot product of raw word counts in either of the blocks. The second is BERT-enhanced TextTiling: vocabulary change is evaluated by comparing the BERT representations of blocks, using cosine similarity. The *nltk* library is used for this [5] - parts of their code also inform the implementation of my own methods.

The embeddings are obtained using RoBERTa (Robustly Optimised BERT pre-training Approach), pre-training configuration of BERT - a fixed size vector representing a sentence is extracted by max pooling of the second to last layer. This baseline and its' implementation is inspired by [35]. Notice that we use RoBERTA in this baseline, but S-BERT in BERTopic. This poses a limitation to our comparison in performance, because some of the difference could be attributed to a difference on fine-tuning BERT. Amending this limitation is left for future work.

The choice of baselines omits certain commonly used segmentation baselines or other state-of-the-art models. This is because this thesis is mostly interested in comparing with similar approaches, to evaluate whether BERTopic can be meaningfully leveraged to improve existing models. Since TextTiling is the choice of framework to modify, we compare only to methods using the same framework. If this modification proves competitive, inserting it into other frameworks for topic segmentation can follow in the future.

3.4 Fitting BERTopic

This section will first introduce in more detail the problem of fitting BERTopic, thereby motivating the analysis to follow. After elaborating on the different approaches to fitting that will be undertaken, we will evaluate the resulting models with respect to several criteria. The analysis will begin with an evaluation and discussion of quantitative measures, including topic coherence, the number of topics found, and the number of

unlabelled segments - some of the measures will require definition before they are evaluated. Finally, the output topics will be qualitatively evaluated by observing specific examples of topics, and critically compared to the numerical measures. This section will be addressing research question (2): How can we best fit document-level topic models to perform well on the downstream application of topic segmentation?

In fitting BERTopic, the primary challenge is deciding on the most appropriate from of documents for the task at hand. The ideal length for the documents is not specified: in the BERTopic paper, they use a variety in different documents for fitting, ranging from treating a tweet as a document, to a BBC article as a document. For the purpose of topic segmentation using TextTiling, the documents that will be useful for us are short segments of the AMI corpus, ranging from 5 to 30 utterances in length. We assume that BERTopic will perform best when fitted on segments of the same length that it will later be transforming into topic distributions: if we were to fit the model to entire meetings, is is unlikely it could find those topics in few-sentence long excerpts of meetings. This assumption is taken from the original BERTopic paper. It is for this reason that we cannot simply fit BERTopic to the full topic-coherent sections found in AMI - these segments will be much longer then segments useful for TextTiling.

Therefore, we will want to segment the AMI corpus into shorter excerpts before fitting. Several approaches to fitting BERTopic will be explored in this section. Some approaches will leverage the topic annotations, but a more naive baseline approach neglecting this information will be also be evaluated here.

Table 3.2 shows an example of a topics' representative words, alongside a "representative document": a document used during fitting, that is close to the representation of that topic in the embedding space. Effectively, this is a segment that is labelled with that particular topic with a high degree of certainty. This table serves to depict to the reader what BERTopic's outputs mean in practice. This is an example from BERTopic fit on segments of 10 sentences: though you may notice there are actually 11 sentences in the representative docs, this is because the sentence "Yeah." was removed in a pre-processing step, due to only containing a word that does not carry much meaning.

We will now delve into the different fitting approaches that will be evaluated:

- **No Fitting** prior to segmentation of a single meeting: this effectively means, BERTopic only fit to one meeting, to the segments that it is transforming into probability distributions to input into the TextTiling pipeline. We do not expect this approach to yield good results: it is only included here as a sort of lower bound.
- **Fixed**: a sliding window with a fixed size of n sentences is moved across the document. At each position, the text in the window is input as a document to fit the model on. Notice that this type of fitting requires no knowledge of data or existing topic annotations: it is a completely unsupervised approach. Therefore this type of fitting will be performed on the test data itself, to mimic the way it would be used in a real-world scenario.
- **Coherent**: this approach also involves a sliding window of size n, with one additional constraint - segments which contain a topic boundary in the gold

Topic Words: battery, recharge, batteries, kinetic, energy, source, rechargeable, charger	
Speaker	**Caption**
C	Well uh f battery , we use uh about uh [disfmarker]
B	Is it n the two AA batteries in it . AA rechargeable batteries .
C	Yeah .
C	Rechargeable of course , because we have the charger .
B	Yeah rechargeable batteries . We have the charger so it's no problem .
C	Yeah and you just [disfmarker]
A	So one one battery ?
C	On uh yeah one battery .

Table 3.2: A topic's representative words are shown in the first row, followed by one of its' representative documents.

standard annotation are discarded, to ensure the segments are more topically coherent. This type of fitting is performed on the training set only, because it involves using information supplied by the annotations.

- **Fixed+Seed**: even segments will be used here, with an addition of the guided variant of BERTopic. The topic seeds correspond to topics in the AMI corpus annotation, and they are hand-crafted based on a manual inspection of the data.

In all of the approaches outlined above, there are two main parameters to vary, that will influence the resulting model. The first is the window (or segment) size. Segments which are too short may result in poor-quality topics: for example, a single utterance is usually topically uninterpretable even for a human reader. On the other hand, segments which are too long may perform poorly on the task of topic segmentation, where shorter blocks are typically considered.

Note that the way we segment our data into documents will also influence the number of documents we are fitting to: if we are using shorter segments, there will be more of them. This brings us to the next varying parameter - stride (or increment). In order to increase the number of segments we are fitting on, especially when fitting to longer segments, we may decrease the stride. However, attempts to implement this strategy resulted in poor-quality topics, as illustrated in Table 3.3, which shows an example of a frequently occurring topic identified when fitting on segments of length 20 with a window stride of 4.

Notice that the concepts linked in this topic: ”trends”, ”spongy” and ”paris”, are not intuitively linked at all to a human reader. The topic is not interpretable. We can think of this as a sort of *overfitting* to the fine-tuning data that is unlikely to generalise well onto other meeting transcripts. Due to the inadequacy of the topics captured when we increase the stride too much, we adopt a 20% overlap as the default. Therefore, when the window size is 5, the stride is 4; when the window size is 10, the stride is 8, etc.

BERTopic fitted with segments of length 20, with stride 4				
Topic 3	trends	important	spongy	fruit
	paris	milan	trend	feel

Table 3.3: The word representation of an incoherent topic, caused by large overlap between segments.

Before proceeding to the result of different fitting experiments, the concept of ”-1” labelled documents will be delineated. As outlined earlier, the -1 topic is represented in the embedding space as the average of all the other topic representations. It is labelled as the -1st topic, instead of the 0-th or 1-st topic, because it is generally not meaningful. An example of a -1 topic, and a document labelled with it is given in table 3.4: notice the topic words include largely uninformative words, like ”maybe”, ”well”, ”n’t”. There are some meaningful words too, because in removing a majority of stop-words and fillers, not many uninformative words are left.

Topic Words: remote, well, think, control, ’nt, design, buttons, know, maybe, use	
Speaker	**Caption**
A	Yeah . Well I [gap] , oh [disfmarker]
B	Yeah . Th they are used to use it when they can see the TV so , I don’t know .
D	Yeah .
A	On the other side , we want to have something new . You know , where
D	Yeah .
A	we want to to have something new and
A	So we I think we should still thinking about it . But maybe [gap] .

Table 3.4: An uninformative ”-1” topic represented in words, and one of its’ representative documents.

For human judgement, the representative document for topic -1 is less interpretable then the representative document given in table 3.2. However, some topical information is discernible: TV is mentioned, and they seem to be discussing new suggestions. However, the words in the -1 topic do not capture this. There will always be short segments of conversational data, where no utterance will hold topical information: so we cannot avoid the appearance of -1 topics. However, -1 topics also sometimes insinuate segments, for which we just do not have a better topic under the currently fitted BERTopic, and yet are meaningful in some capacity - as the one illustrated in 3.4. Therefore, the number of -1 topics will be a factor in evaluating the different approaches to fitting BERTopic.

You may notice the token ”’nt” in table 3.2, too. This illuminates the flaw in the pre-processing steps taken: while classic fillers and stop words were taken out, certain

Method	Segment size	-1 Segments (%)	Number of Topics
No Fitting	5	0.05	25
	10	0.05	31
	20	0.06	39
Fixed	5	0.38	214
	10	0.39	123
	20	0.42	53
Coherent	5	0.36	104
	10	0.33	98
	20	0.30	39
Fixed+Seed	5	0.33	178
	10	0.40	86
	20	0.44	45

Table 3.5: Proportion of -1 labelled segments, and the number of topics found by fitting BERTopic in different ways

aspects of conversational or informal language were not captured, despite several iterations of expanding the list of fillers: this corpus is very large, and the noise present in conversational language is oft unpredictable.

Now that the concept of -1 labelled documents is clear, we evaluate several models on the number of documents with this table. Table 3.5 summarises the proportion of -1-labelled segments in the fitting data and the number of topics found by the model of several fitting experiments.

It can be seen in Table 3.5 that coherent segmentation results in a smaller proportion of -1 topics, and a smaller number of topics found: this is in line with expectations, as we expect many of the segments which contain a boundary to be topically incoherent, and therefore likely to be closer to the average of topics then any one particular topic.

We can additionally observe that longer segments result in fewer topics. This has two possible explanations. The first is the number of segments: when the segments are longer, the stride is bigger, and there are therefore fewer documents to fit on - when the segment length is 5, there should be twice as many segments, compared to segment length 10. The second explanation is, that in shorter segments BERTopic finds more granular topics, which is reflected in the number of topics. It is likely that both factors are at play here.

Introducing a hand-crafted seed can be seen to reduce the number of topics, regardless of the length of segments - we expect this, because seeding is intended to bring the

segments fitted on closer together in the embedding space, resulting in fewer clusters. Interestingly, a hand-crafted seed reduces uncertain topics for shorter segments, but increases it for longer segments - this is unexpected, as the topics we have crafted the seed to are topics which are not nearly as granular as the ones found in 5 sentence segments.

Upon examining the number of topic labels, it becomes apparent that all fitting approaches generate more topics than the 24 distinct topic labels annotated in the AMI corpus. In light of the interpretation of documents proposed by Du et al. (2013) [12] in the Related Work chapter, we can understand that the words in each topically-coherent segment are sampled from its segment-level topic distribution, which is in turn sampled from a global distribution across topics of the entire document. This intuition can aid in comprehending the number of topics identified by BERTopic and why they may be beneficial in this context. Specifically, the distribution of BERTopic topics across segments provides insight into possible segment-level topics, making a higher number of topics potentially useful for topic segmentation.

As the lower-bound baseline for topic segmentation, BERTopic fit on a single meeting will be used. Table 3.5 shows this approach to have the least -1 segments, but this very large difference might mean the topics are less specific and less interpretable. An example of 2 common topics of the fitting data is show in Table 3.6, when fitting on one meeting with segments of size 10 illustrate the nature of these topics. This phenomena is similar to what happened when we increased the overlap between segments by decreasing the stride.

BERTopic fitted on a single meeting from the AMI corpus

Topic 1	find	complex	mainly	teletext	make
	write	add	stuff	use	controls
Topic 2	users	per	cents	fifty	buttons
	kind	remote	cent	seventy	say

Table 3.6: The word representation of the two most common topics, when fitting on a single meeting.

I will further evaluate the different fitting using the standard topic modelling evaluation metrics of topic coherence (TC) and diversity (TD), elaborated on in the Evaluation section of chapter 2. The results of this are shown in table 3.7. For both metrics, a higher value is a better result.

There is research stating that the NPMI coherence measure only holds for classical models, and are less representative of human judgement for neural topic models - instead, they suggest evaluation based on downstream application performance. However, to motivate later experiments, as well as to enrich the analysis of the results, this is performed nevertheless, as the measure still provides insight into the amount of information captured by a topic model [19]. Additionally, it will enable us to evaluate the correlation between topic coherence and performance for topic segmentation.

	Fixed		Coherent		Fixed+Seed	
Segment Length	TC	TD	TC	TD	TC	TD
5	.038	.69	.044	.68	.031	.70
10	.049	.70	.054	.67	.039	.69
20	.092	.66	.121	.72	.084	.68

Table 3.7: The topic coherence (TC) and topic diversity (TD) values of clusters generated by BERTopic for different segment lengths and fitting approaches.

Table 3.7 shows topic diversity to be similar across the different segment lengths and window types. A much larger difference can be observed in topic coherence. Longer segments used for fitting seem to produce more coherent topics: this is in line with expectations for BERTopic. In the evaluation of the two measures in the BERTopic paper, longer documents also produced more coherent topics [15]. Fitting BERTopic to topic-coherent window segments of the meetings yields the most coherent topic out of the presented approaches. Adding a seed decreases the coherence in Table 3.7: this is unexpected and may allude to a worse performance for topic segmentation.

Fixed Window of Length 5				
Topic 1	speech	recognition	voice	recogniser
	n't	technology	okay	speaker
Fixed Window of Length 20				
Topic 1	scroll	wheel	special	push
	push	pushbutons	button	colour

Table 3.8: The word representation of the two most common topics, when fitting with fixed-size segments.

Table 3.8 shows the few most common topics which occur for fixed window segment fitting, as we vary the segment size. I implore you to notice in the table the appearance of uninformative tokens, which are highlighted in red. These mostly seem to occur when fitting the model on shorter segments of the meetings - we can observe a number of them in fitting with a fixed window of 5 sentences, and none when fitting on 20 sentence long segments.

In the rest of this section, the topics obtained by fitting BERTopic with different segmentation approaches will be compared, to highlight a few other effects that the different segmenting approaches have had on the resulting topics. Table 3.9 shows this comparison for segments of length 10.

Before, we showed that seeding reduces the number of topics which are assigned the -1 topic in some cases, and suggested this may be because seeding makes the clusters form

closer together, with a higher degree of certainty, reducing the number of documents that lay in the middle. However, Table 3.9 shows that this reduction came at a price: topic 0 now appears much less informative, so it seems that seeding has added an additional non-informative topic to the list.

However, in the seeded approach, Topic 2 appears to be more informative for the specific domain of interest. It is worth noting that these topics are not randomly ordered, but rather by frequency. In the absence of seeding, Topic 2 pertains to numbers, which is a vague category that provides little insight into higher-level topics. Numbers could be mentioned in a wide range of contexts, from personal life to phone numbers to hardware. However, in the seeded approach, Topic 2 relates to batteries and related concepts, which are more domain-specific and useful for capturing the themes discussed in the AMI meetings.

BERTopic fitted on Fixed Window Segments of Length 10					
Topic 0	remote	think	hold	control	number
	design	appearance	buttons	button	press
Topic 2	two	six	three	four	seven
	five	one	point	say	give
BERTopic fitted on Fixed Window Segments of Length 10, With a Seed					
Topic 0	think	buttons	remote	design	well
	nt	control	use	button	know
Topic 2	battery	batteries	solar	energy	kinetic
	rechargeable	cells	dynamo	power	charger
BERTopic fitted on Topic-Coherent Segments of Length 10					
Topic 0	remote	think	use	control	number
	design	appearance	buttons	button	press
Topic 2	battery	kinetic	batteries	dynamo	energy
	cells	rechargeable	solar	charger	power

Table 3.9: The word representation of the two most common topics, when fitting on a single meeting.

When looking at the topics of Coherent fitting in Table 3.9, we observe there are not many noise words present in the resulting topics, which is expected. Noise words, especially in the first topic might largely come as a result of the difficulties in assigning a topic to segments which are not coherent in topic. They might fall into several topics, making them fall into an inbetween space. Again, the topics are more domain-specific then in fixed window segment fitting. It is not quite clear why this happens, but it might enable better performance for topic segmentation.

This section has given insight and intuition about the kinds of topics that BERTopic

operates with. In terms of coherence, we have seen that longer segments yield an improvement - however, this does not guarantee better performance for topic segmentation. Furthermore, fitting on topic-coherent segments showed an improvement in the topic coherence. This section will therefore serve as a background and motivation for how the final experiments will be approached, and will be referenced in the discussion to make links between topic model evaluation measures and downstream performance.

3.5 TextTiling

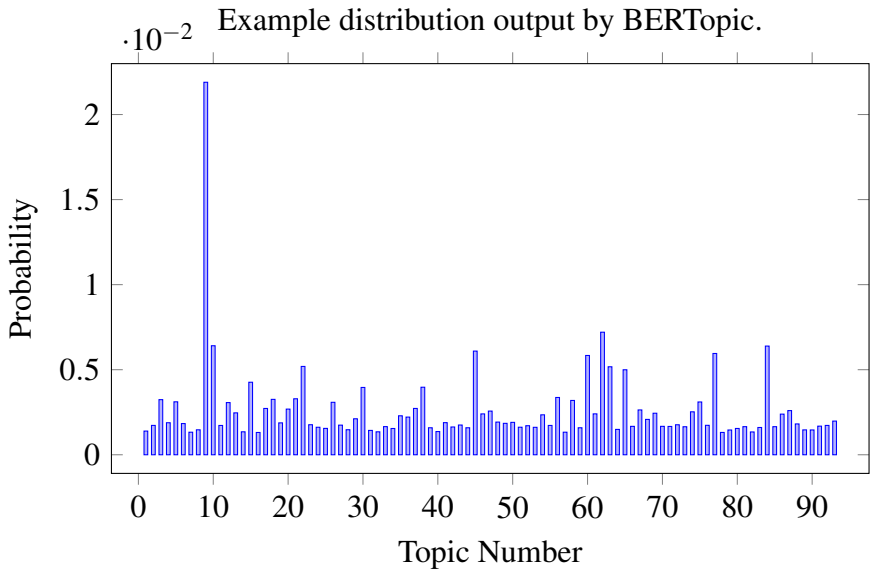
The TextTiling algorithm is based on the idea that a topic change is often indicated by a change in vocabulary. It's intuition and steps are described in more detail in the previous chapter.

This algorithm is easily modifiable by modifying the measure for window similarity: for example, instead of computing similarity with raw word counts, distances between the embeddings of these windows can be used. Using this method, the best performance has been achieved with BERT embeddings [35] [40], where the similarity between blocks was measured using cosine similarity:

$$similarity(x_1, x_2) = \frac{\vec{x_1} \cdot \vec{x_1}}{max(|\vec{x_1}| \cdot |\vec{x_2}|, \varepsilon)})$$

where ε is a very small value whose role is avoiding division by zero for very small vectors. This gives each gap a score between -1 and 1.

An alternate block similarity measure will be proposed here, based on the outputs of BERTopic for the blocks on either side of a gap. BERTopic, as outlined in prior sections, will output the probability distribution across possible topics, for each of the blocks. Thinking of the BERTopic output as a vector and using cosine similarity is not productive: the model is not designed to capture similarity in the semantic space. Instead, we need a measure intended for capturing similarity between probability distributions. An example probability distribution over topics, as output by BERTopic is shown in 3.5 - notice that there are several secondary topics that are given a non-negligible probability. This demonstrates the need for comparing the entire distribution, as opposed to only looking at assigned topics. Additionally, if we consider probabilities instead of topic IDs, we will be able to gain an insight into the content of -1 segments, that topic IDs would not provide.



The similarity measure will be the negative mean of the element-wise Kullback–Leibler (KL) divergence between the probability distributions output by BERTopic:

$$kl\_div(x,y) = \begin{cases} x\log(x/y) - x + y & x > 0, y > 0 \\ y & x = 0, y \geq 0 \\ \infty & otherwise \end{cases}$$

KL divergence is a convenient measure, because big penalties are only accrued when the probability of a certain topic is (relatively) large and the other is not. When both x and y are small for a certain topic, the penalty is small even if they are quite different - topics that are not relevant in either block will therefore not be penalised highly even when they are different.

KL divergence is a statistical measure of distance between two probability distributions. In many contexts, a flaw of KL divergence is that it is not symmetric, so it is not strictly speaking a metric. However, there is no reason why this should be a hindrance in our use. Notice, that since x and y are probabilities, no negative values should occur, so no infinity outputs will be produced. However,where y is near 0, and x is not, very high absolute values may occur - since the distribution of the data is used to decide the cutoff values for boundaries, this can constitute a problem. Based on observing the values, the threshold of 2 is chosen: any value of KL that is higher then this, is replaced by 2. Since KL measures divergence, and we want to measure similarity, we use negative KL divergence - this will be larger (closer to 0) the more alike two segments are to one another.

An outline of BERTopic-encanced TextTiling is show in Figure 3.1. BERTopic transforms each block of sentences (in the diagram the block consists of 2 sentences, for simplicity) into a probability distribution, from which the similarity scores are computed. This is followed by the computation of depth scores and finally placing boundaries - these steps are taken from the original implementation of TextTiling and are explained

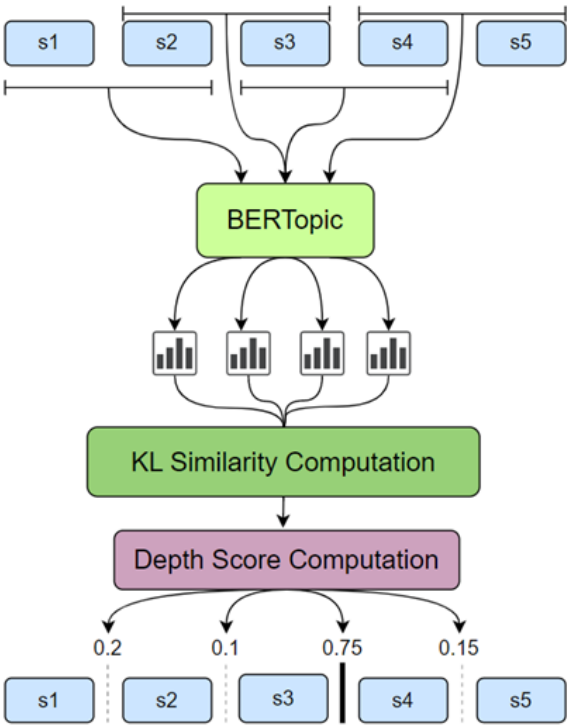


Figure 3.1: Caption

in more detail in 'Background'. The smoothing step of TextTiling is skipped in this diagram for readability, but is present in the implementation of our method.

TextTiling requires several parameters, the meaning of which is detailed here:

- The **cutoff policy**, which governs the number of boundaries, by deciding how deep a valley needs to be to constitute a topic boundary. The cutoff value c is always based on the distribution of depth scores, it is not an absolute value. The cutoff policy decides how that value is calculated from the mean and standard deviation: the default cutoff policy in the *nltk* implementation of TextTiling is $c = mean - \frac{stdev}{2}$. In some of the experiments, the even higher cutoff policy of $c = mean$ was used.
- The **smoothing parameters** influence the effects that the smoothing step has on the data: it is the smoothing window width and the rounds of smoothing that can be manipulated, and avoid local lexical change to result in false boundaries - these parameters differ most for the different experiments. I do not consider the motivation or theory for the choice of these parameters, they are optimised using a dev-set.
- The **window parameter** defines the size of the blocks on each side of a gap that are considered when computing a similarity score. This parameter is one of the most important and will be explored in most depth in this thesis, as it will give insight important for answering the research questions. Specifically, the research question (2): How can we best fit BERTopic to perform well on the downstream task of topic segmentation for conversational data? In classical TextTiling, the

default size for this is 10 pseudo-sentences.

- **Pseudo-sentence size** is the last parameter required by the Texttiling algorithm that I will mention here. In the case of classical TextTiling, it is easy to see why pseudo-sentences are necessary: it operates with raw word counts, which are influenced directly by sentence length. However, BERT-enchanced TextTiling has less of a need for this synthetic segmentation: in fact, the extension of TextTiling with BERT by Solbiati et al. (2021) [35] does not use pseudo-sentences to form blocks, but the actual sentences in the corpus. Following this research, we do the same, and disregard the notion of pseudo-sentences in our extension of TextTiling.

The values of these parameters in different experiments can be found in the Appendix, as they are non-essential to the understanding of my work, but may be of interest to the reader.

This algorithm for segmentation I have chosen to extend is not the state-of-the-art. There are two main reasons why it is the appropriate model for the purpose of this thesis. Firstly, it is an easy framework to modify with alternate source for similarity information. Secondly, it has been enhanced with competitive models (BERT) before, and applied to the corpus we are using [35]. This will allows a good comparison of the methods, and therefore be essential in determining the value that BERTopic can have for the task of topic segmentation.

3.6 Combining Similarity Metrics

In the final experiment, a combined similarity metric will be introduced into the Texttiling pipeline. This approach will use both BERT and BERTopic as sources of information when assessing the similarity between two blocks. This experiment is essential to solidify our answer to the primary research question: can we leverage large pre-trained topic models to improve topic segmentation algorithms? It will clear up whether BERTopic can add information that BERT itself cannot capture. For this experiment, the best performing BERTopic model from prior experiments is used.

First, both methods will separately be used to compute the similarity scores, according to their own metrics. The simple intuition is to just add up the gap scores, however we must be careful not to allow one of the information sources to dominate the other. Note, that cosine similarity is a value between -1 and 1, whereas KL divergence is not bound in the same way - therefore, a normalisation step is necessary before combining the sources. The simple statistical method of score standardisation is used - instead of raw scores, the number of standard deviations by which the raw scores deviate from the mean. The normalised score for each gap is given by:

$$z = \frac{x - \mu}{\sigma}$$

where x is a specific gap score, μ is the mean of that method's gap score, and σ its' standard deviation.

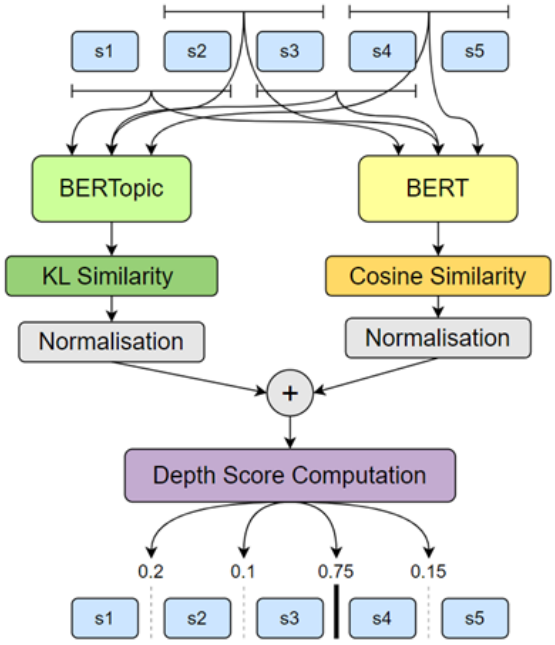


Figure 3.2: Caption

The normalisation step ensures that the similarity scores reside on the same scale when we add them up. This yields the final gap similarity score, that will be given to the rest of the TextTiling pipeline to process.

The full process of the method combining BERT and BERTopic is depicted in Figure 3.2. Again, the blocks are taken to be of size 2 in the diagram, and the smoothing is omitted, to make the diagram clearer.

Chapter 4

Results & Discussion

4.1 Introduction

In this section, we evaluate BERTopic-enhanced TextTiling, as described in 'Methodology', using the differently fit BERTopics that were outlined. The dataset and the evaluation metrics used are described in more detail in sections 3.1 and 2.7, respectively. The results will be explicitly related back to the research questions defined in the Introduction.

This chapter is organized as follows. First we report the results of baselines to provide a point of comparison - this includes using BERTopic with no fitting prior to segmenting single meeting, which serves as a lower bound baseline. We continue by presenting the results of an unsupervised approach, using in-domain data but not leveraging any of its' annotations. This is followed by evaluating the two proposed approaches for adding supervision or guidance to the model. The best performing model will then be combined with the BERT-based model, as detailed in section 3.6. Finally, we will discuss the results and their implications further in section 4.5.

4.2 Baseline Experiments

In this section I evaluate and discuss the baseline models, some of which are my own work. Table 4.1 shows the performance of the baseline models on the training subset of the AMI corpus. The first 4 methods in this table are not my own work - their details and relevant settings are specified in 'Methodology'. These points of comparison are essential for answering the primary research question, since they provide a benchmark we are trying to outperform.

The other four experiments include BERTopic with no fitting prior to segmenting any meeting. We call the method *No Fitting* - note that BERTopic actually does not allow you to run it without fitting, so in more technical terms, the model in this experiment is fit on the segments it is transforming, separately for each meeting. This experiment will motivate further steps undertaken to answer research question: (2) How can we best fit document-level models to perform well on the downstream application of topic

segmentation for conversational corpora?

Algorithm	Pk ↓	WindowDiff ↓
Random	0.60	0.70
Even	0.51	0.55
TextTiling	0.41	0.44
TextTiling+BERT	**0.38**	**0.37**
No Fitting 5	0.44	0.39
No Fitting 10	0.43	0.39
No Fitting 20	0.45	0.42
No Fitting 30	0.45	0.41

Table 4.1: The Pk and WindowDiff values of the baseline experiment on a test subset of the AMI corpus.

The BERTopic experiments in Table 4.1 do not perform well, which is in line with expectations - they serve as a lower bound for comparison. The performance is better then that of the naive baselines of *Even* and *Random*, but worse then the other external baselines, including classical TextTiling - an out-dated model, based on raw word counts.

Poor performance can be attributed to the fact that BERT is trained on written corpora, which makes BERTopic inadequate for this task before we have fit it to in-domain data. Recall the motivation for our approach was that, in contrast with BERT, we can introduce into the model knowledge on global topics. In this approach, we are yet to do that - here, the information captured by BERTopic is a small subset of information that BERT by itself captures. Some topics found when fitting on a single document were shown in 'Methodology' in Table 3.6 - you may remember that the topics were uninterpretable, and linked words that seem unrelated. This is another reason why we did not expect competitive results from this approach.

The poor performance when fitting on a single meeting implies that our approach is not useful when we only have one in-domain meeting - this is a limitation the other baselines in the table do not have. This limitation was clear since the beginning of this research, but is highlighted here anyways for the purpose of clarity.

The results in this section shows, as we have assumed from the start, that the model is not adept for conversational data or topic segmentation by itself. This justifies the need to apply the more intricate fitting approaches introduced in 'Methodology' before we can see BERTopic yield improvements for the TextTiling algorithm.

4.3 Segment Length

Before tackling the two fitting methods that include a level of supervision, the Fixed Window fitting approach is used to deduce the ideal segment length to proceed with. It was highlighted in the section 'Fitting BERTopic', that longer segments yield higher coherence - however, this must be balanced with the needs of the TextTiling algorithm, which has worse performance when the segments are too long.

Table 4.2 shows the results varying with segment size, when we fit BERTopic with Fixed Window segments, overlapping by 20% - since this approach steers clear of using annotations, the fitting is performed on the test set itself, instead of the held-out training set that will be used in later experiments. By illuminating the effect that the segment (or block) size has on TextTiling with a BERTopic based similarity metric, this section of results helps to answer the main research question: Can we use large pre-trained topic models for topic segmentation? The experiments in the next section are informed by the best-performing segment length in this one - in this way, this section is also vital in answering research question (2): How can we best fit document-level topic models to perform well on the downstream task of topic segmentation for conversational corpora?

Segment Length	Pk ↓	WindowDiff ↓
5	0.41	0.42
10	**0.39**	**0.39**
20	0.40	0.43
30	0.42	0.43

Table 4.2: The Pk and WindowDiff values when fixed window fitting, varying in segment length

Table 4.2 shows the results of the Fixed window fitting of BERTopic. We can think of these results as informing a trade-off between the coherence of the found topics and the ideal length of block for TextTiling: while longer segments resulted in more coherent topics, we see in this table that longer segments do not necessarily improve the performance of BERTopic for topic segmentation - considering segments which are too long introduces noise for the topic segmentation task.

The best performing model in Table 4.2 uses 10 sentence segments: this gives the lowest value for both Pk and the Windowdiff evaluation metric. Based on this result, it is with segments of length 10 that we proceed with in the next section, in which we leverage the existing annotations to improve our models performance. exploring the different approaches to fitting BERTopic and the results they yield.

In comparing the results in Tables 4.1 and 4.2, we can see that the best performer achieves results which are better then those of classical TextTiling. The performance is comparable to BERT-enhanced TextTiling, but it is slightly worse.

This approach requires other in-domain data to be available, but no annotations or prior knowledge about the data is necessary. In terms of real-world application, we can envision it in the situation of a corpus like AMI requiring segmentation: meetings of a project from start to finish. In a situation like this, the approach from this section could be used.

4.4 Leveraging Annotations

In this section, we enhance the model by adding the two different ways of leveraging topic annotations for fitting BERTopic. This section answers the research questions: (2) How can we best fit document-level topic models to perform well on the downstream application of topic segmentation for conversational corpora? And (3) how can we leverage existing data, annotated with topic boundaries, to improve topic segmentation algorithms? Informed by the previous section, we use segments of length 10 for both fitting and segmenting in this section.

Fitting Approach	Pk ↓	WindowDiff ↓
Fixed Window	0.39	0.39
Coherent Window	**0.38**	**0.38**
Fixed Window + Seed	0.41	0.44

Table 4.3: TextTiling results which incorporate different types of BERTopic (all on segments of length 10).

Table 4.3 shows the performance of BERTopic-enhanced TextTiling with the different approaches to fitting undertaken.

We can observe in Table 4.3, that introducing a Seed topic list affects the segmentation performance negatively compared with the same fitting data, but with no topic seed list - the first and last models in the Table are fit on the same data, except for the seed. This does not discard the idea that guided topic modelling could help us leverage annotated data to improve topic segmentation, however the approach to creating the seeding list here was too naive to give the model meaningful topic guidance. Recall also the flaw of guided BERTopic, pointed out in 'Background': it assumes that topic seeds, which are short lists of words, and input documents (here segments) can meaningfully be compared in the embedding space. Fitting BERTopic on Coherent window segments yields the best performance of the proposed approaches, coming very close to the performance of the BERT-only method: it is therefore this method that is best-suited to the downstream application of topic segmentation.

Returning to the topic evaluation of the models, as presented in Table 3.7, it is worth noting that the topic-coherent window segment fitting technique identified the most coherent topics in the text. It is this same fitting approach that achieves the best performance on the downstream task of topic segmentation. This alludes to a correlation between topic coherence, and performance for topic segmentation - as long as the

segments are of a sensible size for the TextTiling algorithm, more coherent topics will yield better results.

The two approaches which we are analysing in this section have an obvious limitation: a need for segmented in-domain data, or good prior knowledge about the contents. Again, this is a limitation we were aware of when undertaking this research, and is natural limitation in an approach where the aim is to leverage existing data. As has been outlined before, there are other annotated meeting corpora out there (for example, ICSI), which we could leverage jointly with AMI to create a system for segmenting meetings in similar sectors to AMI.

4.5 Combining Best Models

The best performing model from section 5.3 (segment length 10, coherent window fitting) is combined with the BERT embedding similarity method for TextTiling, answering the primary research question: Can we leverage large pre-trained topic models to improve topic segmentation algorithms? Here we address this question more directly by investigating whether BERTopic captures any segmentation-relevant information, that BERT by itself does not.

Similarity metric	$P_k \downarrow$	WindowDiff $\downarrow$
BERT	0.38	0.37
Coherent Window	0.38	0.38
BERT+Fixed Window	0.37	0.37

Table 4.4: The Pk and WindowDiff values of the third experiment, combining BERT and BERTopic, shown against the performance of the 2 metrics that it is combining.

Table 4.4 shows the results of TextTiling using the combined similarity metric, compared with each of the metrics' performance in isolation. The joint similarity measure improves performance against both. The Pk is reduced by 2.5%, compared with both BERT-only and BERTopic-only similarity measures for TextTiling. Although the difference is not large, this result affirmatively answers the main research question: large pre-trained topic models can be useful for improving topic segmentation algorithms. Since combining it with BERT improves performance (although marginally), we can conclude that BERTopic extracts some segmentation-relevant information from a text that BERT does not.

The size of this improvement may indicate that the models agree on a majority of the boundaries, with only few boundaries where their collaboration makes a difference. To investigate this claim, we measure the Pk between the predicted boundaries of either measure: the value of Pk between the two sets of boundaries is 0.12. This indicates a very high agreement between the two methods, which supports the claim that the similarity between the two sources of information limit the improvement in performance.

4.6 Discussion

Of the two fitting methods which made use of the AMI annotations, the removing of segments containing topic boundaries yields better performance then a human-written seed - the second not only did not improve performance against an approach with no awareness of the annotation, but in fact made it considerably worse. This could just be an indication that my seeding approach was too naive to help with segmentation: in constructing the seed, I only inspected a few topic-labelled segments and did not go in depth. The approach with no human involvement, on the other hand, captured accurate information about annotations across all the data used for fitting. This is not to say that guided topic cannot be a useful way of using prior information about the data to aid topic segmentation, but clearly the prior information must be more in-depth and relevant then just a naive guess. Perhaps someone with good knowledge about the data could produce a more adequate seed topic list. Another possible explanation is the flawed assumption made in designing the guided variant of BERTopic: a word list and a document are expected to be meaningfully comparable in the embedding space.

The performance improvement of the combined (BERTopic+BERT) model against the BERT-only model is quite small. Considering the results BERTopic has been achieving in other downstream applications, as well as on standard topic modelling evaluation metrics, this is a much smaller improvement than we might have expected. There are two main reasons I see for this:

- For the purposes of topic segmentation, the information supplied by BERT and BERTopic might be quite similar: as explained in the 'Background' chapter, the intuition of BERTopic is that is clusters the BERT embeddings of documents (or in our case, dialogue segments), identifies clusters, and assigns topics to each cluster. The topic probabilities output by BERT are therefore a type of cue for where in the semantic space the embedding lays, which is exactly the information carried by BERT. It is true that by segmenting the BERT embedding space based on the documents it is fit on, it captured important additional information about topics, whether that information is actually useful for topic segmentation, is not certain. This theory is validated by the level of agreement between the BERT and BERTopic approach.
- There is information BERTopic captures, that I have neglected in my approach: since topics in BERTopic all have a representation in semantic space, there is a way of representing the relationships or correlations between certain topics - this is to say, the topics captured are not independent from one another. In my approach, taking the topic probabilities, I have discarded information about correlated topics and possibly lost information that could have enhanced the performance. Taking this information into account could be essential to overcoming the previous issue: it would add additional information that BERT itself does not provide.

In finding a way to harness labelled data, a limitation is also imposed on the model: this model is unlikely to work well on out of domain data, where the domain is quite limited. Additionally, the method of fitting which includes topic-coherent segments, required much annotated data to be replicated for a different domain. Although out-of-domain

evaluation was not performed in this research, we are confident that the model presented in this thesis would not perform well: however, there are possible extensions of this work, with domain transfer in mind, that could overcome this.

The approach using fixed window segments of the data has fewer constraints when it comes to real world application, despite needing much in-domain data. Imagine a real-word corpus, similar to the AMI corpus in structure: you have access to transcripts of meetings tracking some project from start to finish, and you want to segment it into topically coherent segments to help with information retrieval. You can fit BERTopic on all the transcripts available, then use the resulting model to aid in topic segmentation. Although we have not shown this approach to improve performance, a more refined approach to it would be likely to yield improvements.

Overall, the results answer affirmatively the primary research question, as they indicate that topic models can be useful for topic segmentation. In terms of the second research question, concerning fitting the model to topic segmentation, it appears that fitting to segments of length 10 sentences is best for this downstream task, where the best approach is the one that maximises coherence - out of the approaches proposed here, coherent window fitting achieved the best coherence, and the best performance for task segmentation. We showed that there are ways to leverage annotated data to improve topic segmentation, by using it to fit BERTopic, giving an answer to research question 3. We did not delve into the question of domain-transfer and how well the model could generalise to different, but similar data: this, including extensions by introducing other meeting corpora to the fitting data, is left for future work.

Chapter 5

Conclusions

This thesis explored the use of large pre-trained topic models for the purpose of topic segmentation, and showed that BERTopic can improve the segmentation results of TextTiling on the AMI corpus: a multi-party conversational corpus. It was also shown to improve slightly the results of a topic segmentation algorithm based on TextTiling and BERT embeddings. The model extended in this research is not a state-the-art model, and was not intended to yield state-of-the-art results. Rather, the aim was to show that large topic models can be useful for the purpose of topic segmentation for conversational language. Additionally, this thesis gives insight into methods for fine-tuning BERTopic to conversational language, as well as fine-tuning it for the downstream task of topic segmentation. It provides both a way to do this with annotated data, as well as a method of performing this with conversational data that has no topic labels.

I see many possible extensions of this as possible future work. As mentioned before, the TextTiling framework I extended in this thesis, is not a state-of-the-art algorithm for topic segmentation, it was only chosen here as ab easy framework to modify, allowing simple comparison with other methods, like BERT cosine similarity. It is therefore straightforward that, since BERTopic shows an improvement against a simple BERT-based method, inserting BERTopic into contemporary topic segmentation frameworks could yield better results, and is worth exploring in the future.

It has been suggested in prior sections, that there is other information that has been shown to boost performance in combination with BERT embeddings, for example, discourse coherence, that could be added to the model to improve it further. Additionally, speaker shifts and speaker roles, which are possible to obtain from automatically generated meeting transcripts, are likely to prove useful in topic segmentation: it is likely that a part of a meeting where speaker B is the primary speaker, and a part f the meeting where speaker B is mostly silent, discuss different topics. The AMI corpus also has the annotations that would enable this exploration, that could likely improve the performance of the model I have proposed.

The BERTopic based topic segmentation algorithm, although only evaluated on in-domain data: so other data in the AMI corpus, can be assumed to be lacking in terms of domain transfer. A quick glance at the tables showing the word representations of some

of the most represented topics will show this: many of the topics found by BERTopic relate very specifically to the topic of the AMI meetings: remote control design. Using data from meetings in different domains in the same fitting produce, might result in a model that is more robust to domain transfer, and could potentially be used to segment sequences of meetings of other sectors or industries, without needing annotated data in that specific sector.

Notice, that we discarded a part of the information carried by BERTopic: namely, the relationships between topics. An additional extension of this simple model would be a topic segmentation algorithm which takes this information into account: this could improve the performance even further. Accounting for related topics would also help to enable another important extension to this work: adding topic labels to the segments identified. This capability of the major strengths of using a topic-modelling driven approach to topic segmentation: because the segments are already based on topical information, the topical information within that segment can easily also be leveraged to find a topic label. For example, the first few words of the most represented topic could constitute a simple, yet likely informative topic label.

Bibliography

[1] Dimitar Angelov. "Top2Vec: Distributed Representations of Topics". In: *ArXiv* abs/2008.09470 (2020).

[2] Sebastian Arnold et al. *SECTOR: A Neural Model for Coherent Topic Segmentation and Classification*. 2019. DOI: 10.48550/ARXIV.1902.04793. URL: https://arxiv.org/abs/1902.04793.

[3] Doug Beeferman, Adam Berger, and John Lafferty. "Text Segmentation Using Exponential Models". In: *Second Conference on Empirical Methods in Natural Language Processing*. 1997. URL: https://aclanthology.org/W97-0304.

[4] Doug Beeferman, Adam L. Berger, and John D. Lafferty. "Statistical Models for Text Segmentation". In: *Machine Learning* 34 (1999), pp. 177–210.

[5] Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.", 2009.

[6] David M. Blei and John D. Lafferty. "Dynamic Topic Models". In: *Proceedings of the 23rd International Conference on Machine Learning*. ICML '06. Pittsburgh, Pennsylvania, USA: Association for Computing Machinery, 2006, pp. 113–120. ISBN: 1595933832. DOI: 10.1145/1143844.1143859. URL: https://doi.org/10.1145/1143844.1143859.

[7] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. "Latent Dirichlet Allocation". In: *J. Mach. Learn. Res.* 3.null (Mar. 2003), pp. 993–1022. ISSN: 1532-4435.

[8] Gerlof J. Bouma. "Normalized (pointwise) mutual information in collocation extraction". In: 2009.

[9] Keyi Cheng et al. *Multi-scale Hybridized Topic Modeling: A Pipeline for Analyzing Unstructured Text Datasets via Topic Modeling*. 2022. DOI: 10.48550/ARXIV.2211.13496. URL: https://arxiv.org/abs/2211.13496.

[10] Jacob Devlin et al. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. 2018. DOI: 10.48550/ARXIV.1810.04805. URL: https://arxiv.org/abs/1810.04805.

[11] Adji B. Dieng, Francisco J. R. Ruiz, and David M. Blei. "Topic Modeling in Embedding Spaces". In: *CoRR* abs/1907.04907 (2019). arXiv: 1907.04907. URL: http://arxiv.org/abs/1907.04907.

[12] Lan Du, Wray Buntine, and Mark Johnson. "Topic Segmentation with a Structured Topic Model". In: *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language*

Technologies. Atlanta, Georgia: Association for Computational Linguistics, June 2013, pp. 190–200. URL: https://aclanthology.org/N13-1019.

[13] Roman Egger and Joanne Yu. "A Topic Modeling Comparison Between LDA, NMF, Top2Vec, and BERTopic to Demystify Twitter Posts". In: *Frontiers in Sociology* 7 (2022). ISSN: 2297-7775. DOI: 10.3389/fsoc.2022.886498. URL: https://www.frontiersin.org/articles/10.3389/fsoc.2022.886498.

[14] Goran Glavaš, Federico Nanni, and Simone Paolo Ponzetto. "Unsupervised Text Segmentation Using Semantic Relatedness Graphs". In: *Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics*. Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 125–130. DOI: 10.18653/v1/S16-2016. URL: https://aclanthology.org/S16-2016.

[15] Maarten Grootendorst. *BERTopic: Neural topic modeling with a class-based TF-IDF procedure*. 2022. DOI: 10.48550/ARXIV.2203.05794. URL: https://arxiv.org/abs/2203.05794.

[16] Marti A. Hearst. "Text Tiling: Segmenting Text into Multi-paragraph Subtopic Passages". In: *Computational Linguistics* 23.1 (1997), pp. 33–64. URL: https://aclanthology.org/J97-1003.

[17] Marti A. Hearst and Christian Plaunt. "Subtopic Structuring for Full-Length Document Access". In: *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '93. Pittsburgh, Pennsylvania, USA: Association for Computing Machinery, 1993, pp. 59–68. ISBN: 0897916050. DOI: 10.1145/160688.160695. URL: https://doi.org/10.1145/160688.160695.

[18] Darell Hendry et al. "Topic Modeling for Customer Service Chats". In: *2021 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*. 2021, pp. 1–6. DOI: 10.1109/ICACSIS53237.2021.9631322.

[19] Alexander Hoyle et al. "Is Automated Topic Model Evaluation Broken? The Incoherence of Coherence". In: *Advances in Neural Information Processing Systems*. Ed. by M. Ranzato et al. Vol. 34. Curran Associates, Inc., 2021, pp. 2018–2033. URL: https://proceedings.neurips.cc/paper_files/paper/2021/file/0f83556a305d789b1d71815e8ea4f4b0-Paper.pdf.

[20] Masaru Isonuma et al. "Tree-Structured Neural Topic Model". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, July 2020, pp. 800–806. DOI: 10.18653/v1/2020.acl-main.73. URL: https://aclanthology.org/2020.acl-main.73.

[21] A. Janin et al. "The ICSI Meeting Corpus". In: *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03).* Vol. 1. 2003, pp. I–I. DOI: 10.1109/ICASSP.2003.1198793.

[22] Taiga Kirihara et al. "Topic Segmentation for Interview Dialogue System". In: *2021 5th International Conference on Natural Language Processing and Information Retrieval (NLPIR)*. NLPIR 2021. Sanya, China: Association for Computing Machinery, 2022, pp. 45–53. ISBN: 9781450387354. DOI: 10.1145/3508230.3508237. URL: https://doi.org/10.1145/3508230.3508237.

[23] Omri Koshorek et al. *Text Segmentation as a Supervised Learning Task*. 2018. DOI: 10.48550/ARXIV.1803.09337. URL: https://arxiv.org/abs/1803.09337.

[24] Jey Han Lau, David Newman, and Timothy Baldwin. "Machine Reading Tea Leaves: Automatically Evaluating Topic Coherence and Topic Model Quality". In: *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*. Gothenburg, Sweden: Association for Computational Linguistics, Apr. 2014, pp. 530–539. DOI: 10.3115/v1/E14-1056. URL: https://aclanthology.org/E14-1056.

[25] Jiwei Li et al. "Adversarial Learning for Neural Dialogue Generation". In: *CoRR* abs/1701.06547 (2017). arXiv: 1701.06547. URL: http://arxiv.org/abs/1701.06547.

[26] Yanran Li et al. "DailyDialog: A Manually Labelled Multi-turn Dialogue Dataset". In: *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Taipei, Taiwan: Asian Federation of Natural Language Processing, Nov. 2017, pp. 986–995. URL: https://aclanthology.org/I17-1099.

[27] Luyang Liu et al. "Neural Variational Correlated Topic Modeling". In: May 2019, pp. 1142–1152. ISBN: 978-1-4503-6674-8. DOI: 10.1145/3308558.3313561.

[28] Y. Liu et al. "Topical Word Embeddings". In: *AAAI Conference on Artificial Intelligence*. 2015.

[29] Iain Mccowan et al. "The AMI meeting corpus". In: *Int'l. Conf. on Methods and Techniques in Behavioral Research* (Jan. 2005).

[30] Hemant Misra et al. "Text segmentation via topic modeling: an analytical study". In: *Proceedings of the 18th ACM conference on Information and knowledge management* (2009).

[31] Dat Quoc Nguyen et al. "Improving Topic Models with Latent Feature Word Representations". In: *Transactions of the Association for Computational Linguistics* 3 (2015), pp. 299–313. DOI: 10.1162/tacl_a_00140. URL: https://aclanthology.org/Q15-1022.

[32] Lev Pevzner and Marti A. Hearst. "A Critique and Improvement of an Evaluation Metric for Text Segmentation". In: *Computational Linguistics* 28.1 (2002), pp. 19–36. DOI: 10.1162/089120102317341756. URL: https://aclanthology.org/J02-1002.

[33] Martin Riedl and Chris Biemann. "TopicTiling: A Text Segmentation Algorithm based on LDA". In: *Proceedings of ACL 2012 Student Research Workshop*. Jeju Island, Korea: Association for Computational Linguistics, July 2012, pp. 37–42. URL: https://aclanthology.org/W12-3307.

[34] Min Shi et al. "WE-LDA: A Word Embeddings Augmented LDA Model for Web Services Clustering". In: *2017 IEEE International Conference on Web Services (ICWS)* (2017), pp. 9–16.

[35] Alessandro Solbiati et al. *Unsupervised Topic Segmentation of Meetings with BERT Embeddings*. 2021. DOI: 10.48550/ARXIV.2106.12978. URL: https://arxiv.org/abs/2106.12978.

[36] Ryuichi Takanobu et al. "A Weakly Supervised Method for Topic Segmentation and Labeling in Goal-oriented Dialogues via Reinforcement Learning". In:

Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18. International Joint Conferences on Artificial Intelligence Organization, July 2018, pp. 4403–4410. DOI: 10.24963/ijcai.2018/612. URL: https://doi.org/10.24963/ijcai.2018/612.

[37] Liang Wang et al. "Learning to Rank Semantic Coherence for Topic Segmentation". In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Copenhagen, Denmark: Association for Computational Linguistics, Sept. 2017, pp. 1340–1344. DOI: 10.18653/v1/D17-1139. URL: https://aclanthology.org/D17-1139.

[38] Wen Xiao and Giuseppe Carenini. "Extractive Summarization of Long Documents by Combining Global and Local Context". In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 3011–3021. DOI: 10.18653/v1/D19-1298. URL: https://aclanthology.org/D19-1298.

[39] Linzi Xing and Giuseppe Carenini. *Improving Unsupervised Dialogue Topic Segmentation with Utterance-Pair Coherence Scoring*. 2021. DOI: 10.48550/ARXIV.2106.06719. URL: https://arxiv.org/abs/2106.06719.

[40] Linzi Xing et al. *Improving Context Modeling in Neural Topic Segmentation*. 2020. DOI: 10.48550/ARXIV.2010.03138. URL: https://arxiv.org/abs/2010.03138.

[41] Yi Xu, Hai Zhao, and Zhuosheng Zhang. "Topic-Aware Multi-turn Dialogue Modeling". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 35.16 (May 2021), pp. 14176–14184. DOI: 10.1609/aaai.v35i16.17668. URL: https://ojs.aaai.org/index.php/AAAI/article/view/17668.

[42] Leilan Zhang and Qiang Zhou. "Topic segmentation for dialogue stream". In: *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE. 2019, pp. 1036–1043.

Appendix A

Parameters

A.1 TextTiling Parameters

Table A.1 specifies the TextTiling parameters in each experiments. When using BERTopic, the parameters were the same for all experiments which used the same segment length, hence their grouping in the table.

Model	Cutoff	Smoothing Width	Smoothing Rounds	Window
BERTopic, length 5	μ	5	2	
BERTopic, length 10	μ	5	2	
BERTopic, length 20	μ	2	2	
BERT	μ	5	2	10
BERTopic+BERT	μ	5	5	10
Classical	μ	10	2	10

Table A.1: TextTiling Parameters in different experiments undertaken.