BeamSeg: a Joint Model for Multi-Document Segmentation and Topic Identification

Pedro José dos Reis Mota

CMU-LTI-19-007

Language Technologies Institute
School of Computer Science
Carnegie Mellon University
5000 Forbes Ave., Pittsburgh, PA 15213
www.lti.cs.cmu.edu

Thesis Committee:

Anselmo Penas
Chris Dyer
Robert Frederking
Bruno Emanuel da Graça Martins

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
In Language and Information Technologies

Section II-B. Rights pursuant to Open Access Publishing Plus

Author's election of Open Access as the Type of Publishing confirms Author's choice to have ProQuest publish the Work according to the Open Access Publishing option described here.

Open Access Publishing Plus. In addition to the rights granted under Section I of this ProQuest Publishing Agreement, ProQuest may reproduce, distribute, display and transmit the Work in electronic format in the ProQuest Dissertations & Theses database, where it may be made available for free download. A subset of the ProQuest Dissertations &Theses database, currently known as PQDT Open, may be accessed by the academic community as well as through major search engines and open access harvesters. ProQuest may also provide an electronic copy of the Work to Author's degree-granting institution where it may also be posted for free open access. Learn more: http://www.proquest.com/en-US/products/dissertations/epoa.shtml

Copy Sales. ProQuest and its agents and distributors may offer copies of the Work for sale in tangible media, including but not limited to microform, print and CD-ROM, as well as in electronic format either individually or as part of its electronic database and reference products and services. No royalties shall be due to Author.

Publishing Fees. Author's payment of the additional Open Access fee is a one-time, up-front fee in addition to the ProQuest dissertation or thesis publishing fee. Author's institution may assess additional fees to be collected along with the Open Access and publishing fees.

Section III. Publishing Options & Signature

Select the publishing options below that best fit your interests and scholarly publishing obligations.

Traditional Publishing

☒ I want to make my work widely available and I want to be eligible to receive royalties on the sale of my work.

- I understand that I must maintain a current mailing address with ProQuest in order to be eligible to receive royalties.
- I understand that the ProQuest **fee for Traditional Publishing is $55 for Master's thesis and $65 for Doctoral dissertations**, and that my graduate institution may pay all or a portion of the total fee as well as may require additional fees in association with my submission to ProQuest.

Open Access Publishing Plus

□ I want the broadest possible dissemination of my work, and I want to provide free global access to the electronic copy of my work via the internet.

- I understand that I will not be eligible to receive royalties.
- I understand that the ProQuest **fee for Open Access Publishing Plus of Master's thesis is $150 and for Dissertations is $160**, and that my graduate institution may pay all or a portion of the total fee as well as may require additional fees in association with my submission to ProQuest.

SELECT PUBLISHING OPTIONS

I want my work to be available as soon as it is published.

☒ Yes
□ No – I would like access to the full text of my Work to be delayed for the following period of time:
□ 6 month embargo
□ 1 year embargo
□ 2 year embargo

Note: Most institutions have delayed release (embargo) policies, please consult with your Graduate School/Program, if you need to delay the release of your Work. Access to the full-text of your work will be delayed for the time period specified above, beginning from the date that we receive your manuscript at ProQuest. During this time, only your citation and abstract will appear in the ProQuest Dissertations & Theses database (PQDT).

I want major search engines (e.g. Google, Yahoo) to discover my work. Learn more: http://www.proquest.com/en-US/products/dissertations/google.shtml

☒ Yes
□ No

Acknowledgment: I have read, understand and agree to this ProQuest Publishing Agreement, including all rights and restrictions included within the publishing option chosen by me as indicated above.

REQUIRED Author's signature Pedro José dos Reis Mot **Date** 08/19/19

(Print Name) PEDRO JOSÉ DOS REIS MOTA

Institution conferring degree Carnegie Mellon University

This page must accompany your manuscript and the rest of the submission materials.

Please check type of manuscript:

___ M (Master's Thesis)

x D (Dissertation)

Dissertation/Master's Thesis Submission Form

Please print clearly in block letters

Personal Information

Last Name DOS REIS MOTA

Middle Name or Initial

First Name PEDRO JOSÉ

Country (ies) of Citizenship PORTUGAL

Degree & Dissertation Information

Title of Dissertation/Thesis BEAMSEG: A JOINT MODEL FOR MULTI-DOCUMENT SEGMENTATION AND TOPIC IDENTIFICATION

Institution conferring degree CARNEGIE MELLON UNIVERSITY

Degree awarded (abbreviate; e.g., Ph.D.) PH. D.

College, School, or Division LANGUAGE TECHNOLOGIES INSTITUTE

Year degree awarded 2019

Department or Program CMU PORTUGAL

Year manuscript completed 2019

Advisor/Committee Chair MAXINE ESKENAZI, MARIA LUÍSA TORRES RIBEIRO MARQUES DA SILVA COHEUR

Committee Member ANSELMO PENAS

Committee Member BRUNO EMANUEL DA GRAÇA MARTINS

Committee Member CHRIS DYER

Committee Member

Committee Member ROBERT FREDERKING

Committee Member

Language of manuscript ENGLISH

Primary Subject Category: Enter the 4-digit code and category name from Guide 2 that most closely describes the disciplinary area of your research.

Code 0984 Category COMPUTER SCIENCE

You may suggest two additional subject categories that may aid in the discovery of your work in our digital database.

Code________ Category________ Code________ Category________

Provide up to 6 keywords or short phrases for citation indices, library cataloging, and database searching.

TEXT SEGMENTATION

BAYESIAN METHODS

TOPIC IDENTIFICATION

Current Contact Information

Current Email Address pedro.jose.mota@gmail.com

Street Address (line 1) AVENIDA DOUTOR JOSÉ EDUARDO VITOR NEVES, 35A 2D

Street Address (line 2)

City ENTRONCAMENTO

State/Province SANTARÉM

Daytime Phone +351 963785488

Country PORTUGAL

Postal Code 2330-066

Evening Phone +351 963785488

Permanent Contact Information

Permanent Email Address pedro.jose.mota@gmail.com

Street Address (line 1) AVENIDA DOUTOR JOSÉ EDUARDO VITOR NEVES, 35A 2D

Street Address (line 2)

City ENTRONCAMENTO

State/Province SANTARÉM

Future Phone +351 963785488

Country PORTUGAL

Postal/ZIP code 2330-066

Alternate Future Phone +351 963785488

THIS PAGE MUST ACCOMPANY YOUR MANUSCRIPT AND THE REST OF YOUR SUBMISSION MATERIALS

Attach additional, separate copies of your Title Page and Abstract to this form

TÉCNICO LISBOA

Carnegie Mellon University

BeamSeg: a Joint Model for Multi-Document Segmentation and Topic Identification

Pedro José dos Reis Mota

Ph.D. Thesis

Thesis Advisory Committee

Advisors:	Doctor Maxine Eskenazi Doctor Maria Luísa Torres Ribeiro Marques da Silva Coheur
Jury:	Doctor Anselmo Peñas Doctor Chris Dyer Doctor Robert Frederking Doctor Bruno Emanuel da Graça Martins

July 2019

Abstract

The work in this thesis is motivated by the problem of navigating the content of a collection of related documents, which is cumbersome if only a list of documents is given. Automatically structuring the content organization of a dataset by identifying topically cohesive segments and linking segments describing the same topic addresses this issue. Previous work deals with this problem by using a multi-document joint model for segmentation and topic identification at the dataset level, a perspective we also take. This multi-document approach to segmentation contrasts with approaches that segment documents individually. The advantage of a multi-document model is that segmentation is leveraged by repeated descriptions of the same topic across different documents. We continue this line of work by hypothesizing that vocabulary relationships between different segments can be used to obtain a more accurate segmentation and topic segment identification. We also hypothesize that documents that share the same modality (video transcripts, PowerPoint, *etc.*) have similar characteristics that could be modeled to obtain a better performance in these tasks. To study the previous hypothesis, we propose BeamSeg, a joint model for multi-document segmentation and topic identification where it is assumed that segments have vocabulary usage relationships. BeamSeg implements segmentation and topic identification in an unsupervised Bayesian setting by drawing from the same multinomial language model segments with the same topic. Contrary to previous work, we assume that language models are not independent since the vocabulary changes in consecutive segments are expected to be smooth and not abrupt. We achieve this by putting a dynamic Dirichlet prior over the language models that takes into account data contributions from other topics. Additionally, we encode in BeamSeg that documents with different modalities have similar segment length characteristics, and, thus, each modality has its segment length prior. To better understand the performance advantages of the proposed joint model approach, we compare BeamSeg to a pipeline approach (performing segmentation and topic identification sequentially). In this context, we extend two single-document models to the multi-document case and propose a graph-community detection approach to topic identification. In order to test our hypothesis, we carry out a data collection task, as datasets from previous works have few documents with short segments, leaving little room to observe vocabulary relationships. The evaluation using the collected dataset shows that BeamSeg obtains the best results affording this way practical improvements in both segmentation and topic identification and corroborating our hypothesis.

Acknowledgments

Ph.D. journeys seem to always be an odyssey through a winding path with many ups and downs. I am very thankful to have the guidance of two exceptional advisors, Maxine Eskenazi and Luísa Coheur, during this endeavor. Maxine, thank you for putting all your knowledge and research experience into this thesis and for making me strive always to give the best of me. Luísa, thank you for your support in critical moments and your contagious enthusiasm for scientific research. The advice I got from both of you goes well beyond our technical research discussions, which is truly inspiring and a role model for me to follow. For all this, I will always be thankful to you.

I want to thank you all the members of the thesis committee, Chris Dyer, Bruno Martins, Robert Frederking, and Anselmo Peñas. Your feedback and advice regarding this work was essential and allowed me to stay focused on the research line I had to pursue. Chris, thank you so much for taking the time to meet with me and providing further suggestions that definitely improved this work.

To my dear friend Hugo Rodrigues a huge thank you. I am so glad that such a wonderful person enrolled in the same year and the same Ph.D. program as me. Throughout these years we build a friendship for life. Your friendship has truly been incredible support through good and bad times. I will always be amazed when remembering our whiteboard discussion where you put such a massive effort to help me just out of pure kindness. I really cannot thank you enough.

A special thank you to my parents, without their love and sacrifices none of this would be possible. They have always been my main source of inspiration, and I hope I made them proud.

Index

1 Introduction

1.1 Motivation

Documents exhibit an implicit content organization that aggregates related text passages in topically coherent segments. This content organization emerges by placing boundaries in a document where topic shifts occur. The text between consecutive boundaries corresponds to a topic segment. Understanding this document structure enables efficient content navigation. This has become more relevant with the number of available documents in the Web. The current information landscape has also enabled access to documents describing the same subject, providing alternative views or complementary information. This is advantageous in a variety of scenarios. For example, students have at their disposal several learning materials from different modalities (video lectures, textbooks, *etc.*), and might need to find a particular segment that best suits their learning needs. Finding such documents is an easy task since search engines are capable of returning documents conveying related information. However, if search engines are effective in retrieving these documents, the task of putting them into a coherent picture remains a challenge (Shahaf et al., 2012). The research topic of this thesis, automatically finding document segments – *text segmentation* – and identifying which ones discuss the same topic – *topic identification* – addresses this challenge.

Text segmentation approaches rely on the lexical cohesion theory (Halliday and Hasan, 1976), which postulates that discourse structure is correlated with the use of cohesive vocabulary. Thus, topic segments can be identified by detecting vocabulary changes. Research in text segmentation traditionally uses the content of isolated documents to recover topic boundaries (the *single-document* approach), despite the better results that can be obtained when considering segmentation as a global phenomenon in a collection documents (the *multi-document* approach) (Jeong and Titov, 2010). The intuition is that repeated discussions of a topic (Figure 1.1), in different documents, better define it, and, consequently, leverage the segmentation. This might not be so relevant when topic boundaries are clear-cut, such as in news broadcasts, but it is relevant for documents with tightly related segments, where an overcharging topic is described. For example, in the scenario of learning materials, segmenting one document might be a complicated task, as abrupt changes in the vocabulary are not frequent. Thus, looking at different documents can help identifying the different topics. This also makes an argument for text segmentation and topic identification to be jointly modeled since they are related problems. Jeong and Titov (2010) proposed such a joint modeling approach, but some shortcomings exist. One of the problems is that their joint model assumes that repeated topics exist in a vacuum, disregarding interactions between segments chained together to describe an overarching topic. Another shortcoming is the assumption that all documents have the same modality while different modalities have particular segment length properties. For example, we expect that Power Point (PPT) documents to have shorter segments than video lectures. In this thesis, we study how these properties manifest in a collection of related documents and if they can be used to improve text segmentation and topic identification.

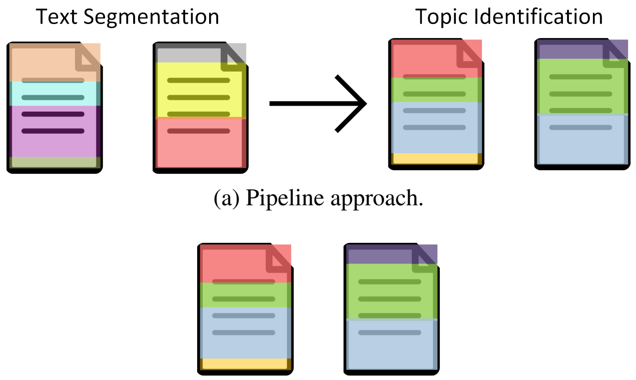
Just as we introduced **average velocity** we now describe **average acceleration**. When **velocity** changes ... over **time**. ... introduce an **average acceleration** ... The **average acceleration** between **time** t2 minus t1. The dimension are ... secs per **time** squared.	**Acceleration** We say ... changing **velocity** are “accelerating” **Acceleration** is the “Rate of change of **velocity**”. You hit the accelerator to speed up (It’s true you also hit ... friction is slowing ...) **Average acceleration** Unit of **acceleration**: m/s2 Meters per second squared	The **acceleration** of a particle ... rate of change of **velocity** ... **time**. **Average acceleration** ... is v2-v1 t2-t1. **Acceleration** may be positive, negative or zero. Zero **acceleration** means we have constant **velocity**... direction and the **acceleration** need not coincide.

Figure 1.1: Examples of segment excerpts from video, slide presentation, and PDF documents, respectively, in the acceleration topic. Words in bold depict the shared vocabulary across segments.

1.2 Thesis Statement

Most approaches consider segmentation at the single-document level. This is a narrow perspective of the segmentation phenomenon since it does not allow to study how different documents describing the same subject structure their topic segments. Another consequence is that if we want to identify the topics between segments in different documents it is necessary to have another algorithm that performs this task *a posteriori* – *pipeline* approach. This is depicted in Figure 1.2a where from the result of the (single-document) text segmentation step it is not possible to deduce the topic relations (all colors are different).

In this thesis, we take the multi-document segmentation stance instead, which segments a collection of documents by leveraging topic segment repetitions in the dataset. This perspective entails that topic identification at the segment level needs to be performed since we need to know which segments share similar lexical cohesion properties – *joint model* approach. This is shown in Figure 1.2b where we can see that the model outputs a topic segment structure in a single step. With this approach, we can study the hypothesis that segmentation can be improved by assuming that lexical cohesion is similar across segments that describe the same topic in different documents. Therefore, we also hypothesize that this joint approach to the multi-document segmentation and topic identification is beneficial.

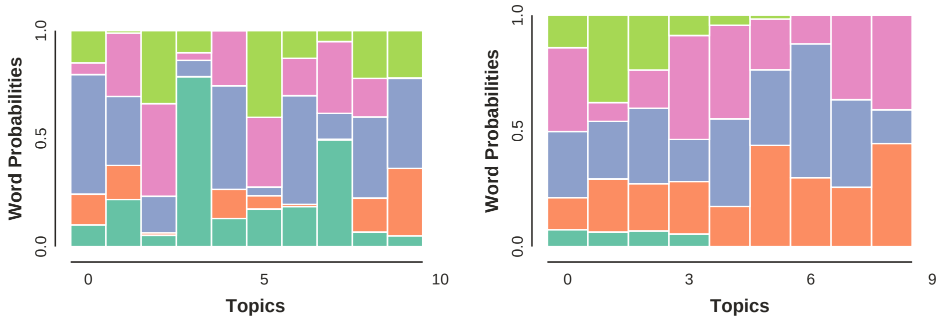


(a) Pipeline approach.

(b) Joint model approach.

Figure 1.2: Different approaches to the tasks of text segmentation and topic identification. Changes in color correspond to segment boundaries. Segments with a matching color describe the same topic

Setting up our work in a joint approach allows us to study novel research paths. A path we also explore is the hypothesis that vocabulary usage relationships between segments are similar across documents describing the same topic. For example, if some word is heavily used in one segment, it is likely that it continues to appear in the following one, though less frequently. We illustrate this in Figure 1.3, where each vertical bar is a segments language model distribution. If there is no vocabulary relationship the language models are random (Figure 1.3a) but what we want to explore is the assumption that some underlying structure exists (Figure 1.3b). Modeling such interactions can improve topic segmentation algorithms. This assumption has been studied for the single-document approach (Eisenstein, 2009; Du et al., 2013), but whether it holds for multi-document segmentation remains an open research question. Even if lexical cohesion is observed across documents, there might be constraints related to the modality of the document that make the segmentation task harder. For example, segments from video transcripts are longer than segments from a slide presentation. Determining how to properly deal with such constraints is also a goal of this thesis.



(a) Topics generated independently.

(b) Topics generated by a dynamic prior.

Figure 1.3: In the figures above the different colors in the bar graph represent word distributions of segment language models (topics). Thus, each color represents a word in the vocabulary. The height of the bars depicts the corresponding word probability.

Given the previous context, we formulate the overarching research hypothesis of this thesis as follows:

- If multi-document segmentation and topic identification are jointly modelled, then the tasks can be more accurately performed.

- If we model segment vocabulary usage relationships and segment length properties based on document modality, then multi-document segmentation and topic identification can be more accurately performed.

The first difficulty in studying our hypothesis is the lack of evaluation resources since multi-document segmentation is an under-explored research area. To overcome this limitation, we collect data where multi-document segmentation can be observed at a larger scale and in different document modalities. We also study how human judges annotate the dataset. By building an appropriate evaluation framework more solid conclusions can be drawn about the progress of proposed techniques and a more reliable measure of progress in the research area can be achieved. Ultimately, this leads to a better understanding of underlying the natural language phenomena that occurs when structuring discourse in topically coherent segments.

To better determine if our first research hypothesis holds, we compare a joint model approach to multi-document segmentation and topic identification to a pipeline approach. Such approaches do not exist in a multi-document setting. Therefore, we extend two existing single-document models to afford multi-document segmentation. To enable topic identification, we propose a graph-community detection approach that takes a segmentation as input and determines which segments share the same topic.

We study our second research hypothesis we propose BeamSeg, an unsupervised Bayesian joint model for text segmentation and topic identification. By using a probabilistic approach, we encode assumptions on how input data was generated. In this context, BeamSeg implements lexical cohesion by assuming that good segmentations spread their probability mass on a restrict set of words. In the probabilistic setting, segmentation is cast as the problem of finding the latent variables parameter configuration that best explains the observed data. The BeamSeg model assumes that each sentence has a latent topic assignment. In turn, topics correspond to word distribution language models. Segments emerge by having consecutive sentences with the same topic. To afford multi-document segmentation, the model assumes that topics can be shared across documents. This topic sharing aspect lends the generative process to a joint model for multi-document segmentation and topic identification. Thus, topic identification is performed by realizing which segments are assigned to the same topic. Following Bayesian approaches, in BeamSeg we assign priors to the language models word distributions. By conditioning the priors on modality and assuming that language models are not independently drawn from the prior, we study how document modality and segment vocabulary relationships impact multi-document segmentation and topic identification. We compare BeamSeg with existing single and multi-document joint models as well as a pipeline approach with our model extensions and graph-community detection algorithm in the collected dataset, contributing this way to answer the raised scientific questions, and, consequently, advance the state-of-the-art in this research area.

From the previous research context, we highlight the following contributions resulting from this thesis:

- A dataset suitable for evaluation of multi-document segmentation at a large scale (Mota et al., 2018b).
- Improvement of single-document models results by extending them to the multi-document case (Mota et al., 2016).
- Improvement of results obtained in a pipeline approach to topic identification using the proposed graph-community detection algorithm (Mota et al., 2018a).
- Improvement of results obtained by state-of-the-art segmentation algorithm using the proposed joint model of segmentation a topic identification that takes into account segment vocabulary relationships and segment length properties.

1.3 Thesis Overview

The remaining of this thesis is structured as follows:

- Chapter 2 provides the research context for this work. It focuses on describing the linguistic perspective of segmentation-related phenomenon and how algorithms can use its constructs to obtain accurate segmentations. Then, we describe the two types of approaches to the segmentation task: lexical similarity-based and probabilistic approaches.
- Chapter 3 discusses the limitations of the available segmentation and topic identification datasets. To address such limitations we carried out a data collection task and performed an annotator agreement study on the collected data.
- Chapter 4 describes the novel work developed in this thesis. It is comprised of three parts: extending existing single-document models to multi-document, a graph-community detection approach to topic identification, and a multi-document joint model for segmentation and topic identification (the main focus of the thesis).

- Chapter 5 describes the segmentation experiments where we determine if the multi-document approach performs better than the single-document approach. In this process, we also investigate how different models are able to adapt to datasets containing different document modalities.

- Chapter 6 describes the topic identification experiments where we compare pipeline strategies (segmentation and topic identification tasks decoupled) and joint model approaches (segmentation and topic identification tasks performed simultaneously).

- Chapter 7 provides a summary of the contributions of this thesis as well as possible future research directions to be explored.

2 Research Context

The design of systems to recover document segmentation boundaries is grounded in linguistic theory. In this chapter, we provide an overview of such theories and how they relate to segmentation (Section 2.1). Then, we survey the literature and describe existing approaches to text segmentation (Section 2.2).

2.1 Lexical Cohesion Theory

An overarching pillar of segmentation algorithms, regardless of implementation, are the linguistic constructs in cohesion theory (Halliday and Hasan, 1976). This theory postulates that grammatical and lexical links within sentences hold a text together and give it meaning, differentiating coherent text from a random set of sentences. From a grammatical point of view, cohesion is achieved through devices of reference, substitution, ellipsis, and conjunction. At the lexical level, cohesion is formed by means of word repetition. By using lexical cohesion properties, algorithms can detect segment boundaries since the use of a consistent vocabulary indicates topic continuity and changes in the vocabulary indicate that a new topic has started.

To demonstrate lexical cohesion in a segmentation scenario, we provide an example in Figure 2.1 showing two video transcript segments from a Physics class. Segment lexical cohesion is observed in the words that characterize segments. For example, the words ‘speed’ and ‘instantaneous’ are used frequently and exclusively in their corresponding segments. This makes sense since the segments describe the ‘average speed’ and ‘instantaneous velocity’ topics. Another interesting word repetition behavior can be observed in the word ‘velocity’, which is used in both segments but more frequently in the second one. This suggests that words usage changes smoothly rather than abruptly, which motivates our approach to model vocabulary usage relationships between topics. This is further corroborated when looking at the full document since we can observe that previous segments do not use this word at all and later segments show a frequency decrease.

The lexical cohesion properties of segments can be used to automatically recover segment boundaries by finding topically relevant words that frequently occur in a continuous span of text. Despite being a relatively simple and intuitive principle, in practice, it is powerful, which explains why it is at the core of all

There is a very big difference in physics between **speed** and **velocity**. The average **velocity** between time t_1 and t_5 is zero but the average **speed** is not. The average **speed** is defined as the distance traveled divided by the time that it takes to travel that distance. Now, what is the distance that the object traveled between time t_1 and time t_5? Well, the object started at a position here on this x axis and then it went up, reached the highest point here so I'll make a drawing for you here. It reached the highest point here, then it went down. And then when it went here it went up again and comes down again and it 's back. And in order to find the average **speed** you would now have to know exactly what this distance is add up this distance add up this distance and this distance. And if that distance altogether were, for instance, 300 meters and if the time between t_1 and t_5 were three seconds then the average **speed** would be 300 meters divided by three seconds. That would be 100 meters per second so the average **speed** would be 100 meters per second yet the average **velocity** would be zero.

==

If you look at the location t_3 and t_2 and I bring t_3 closer and closer to t_2 then this angle of alpha will increase and I can go to the extreme that I bring t_3 almost right at t_2. The angle of alpha will then be tangential to this point. This will then be my angle of alpha. And now you will understand how we define the **instantaneous velocity** at time t which is different from an average **velocity** between two time intervals. The **instantaneous velocity**, v and I pick a random time, t equals the limiting case for x measured at time t plus delta t minus x measured at time t divided by delta t and I do that for delta t goes to zero. ... If it is negative, however, when you're here then it is a negative **velocity**. And if the angle of alpha is zero then the **velocity** is zero. So if we now look at this plot we can search for the times that the **velocity** is zero so you have to look for the derivative being zero. That means the angle alpha being zero. Clearly, here the **velocity** is zero. So those are the times that the **velocity** is zero. What are the times that the **velocity** is positive? Well, it's positive here. The **velocity**'s positive here still positive, positive, becomes negative negative, positive, zero, negative. So that's the definition of v, **instantaneous velocity**.

Figure 2.1: Lexical cohesion example in two segments. Highlighted words depict word repetition behaviors patterns that characterize the segments.

segmentation approaches. In addition to lexical cohesion, grammatical properties of texts can be used in the context of a segmentation task. For example, anaphors preserve topic continuity, because the same object is being referred. Also, some particular lexical items and cue words can be indicative of segment boundaries, since they tend to signal references, substitutions, and conjunctions. These can be identified during the preprocessing stage and later be used as features in segmentation algorithms. In this thesis, we focus on the syntactic aspect of the lexical cohesion theory since it better suits the fully unsupervised proposed approach.

2.2 Text Segmentation

Following the lexical cohesion theory, text segmentation algorithms work by identifying spans of text where prominent changes in vocabulary occur. The main difference between segmentation methods is how lexical cohesion is implemented: some resort to lexical similarity (Section 2.2.1); the remaining follow a probabilistic approach (Section 2.2.2).

2.2.1 Lexical Similarity Approaches

A common way to address natural language processing tasks is through the notion of entity similarity. For a document segmentation task, entities typically correspond to consecutive utterances in a document with a word count vector representation. However, other text units can also be considered, such as paragraphs. The similarity of two utterances u_1 and u_2 is then measured using the cosine similarity function:

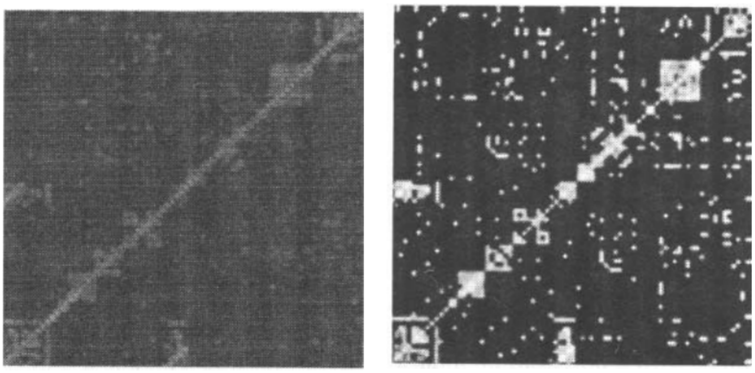
$$S(u_1, u_2) = \frac{u_1 \cdot u_2}{||u_1|| \times ||u_2||}, \tag{2.1}$$

where $u_1 \cdot u_2$ is the dot product of the utterances vector representations and $||u_i||$ is the L_2 norm of u_i. Following lexical cohesion, lexical similarity approaches assume that the vocabulary distribution of segments is homogeneous. Then, using the previous text similarity definition, segments are discovered by finding high similarity regions of documents. The algorithms we describe below follow this lexical similarity approach in a single-document segmentation context.

A classic method using the above setup is TextTiling (Hearst, 1997), which assumes that segment boundaries are found when consecutive sentences have a low similarity value. This is determined not only by the similarity value itself but also by its tendency to increase or decrease when examining the similarity plot of all utterances in a document. Segments are then identified by finding the minimum values in the plot. An empirically defined cutoff threshold tunes the granularity of the segmentation. The threshold specifies the minimum similarity value for which segmentation boundaries are accepted. The evaluation of TextTiling was carried out in articles from science magazines. A 71 and 59% precision and recall scores were obtained, improving a random baseline by 21 and 8%, respectively.

Several other works built on the TextTiling approach (Galley et al., 2003; Balagopalan et al., 2012; Shah et al., 2015). For example, LCseg (Galley et al., 2003) uses the same algorithm but weights the utterance vectors. The weights are computed based on the notion of lexical chains. A lexical chain exists for each word in the vocabulary and is constructed to consist of all repetitions ranging from the first to the last appearance of the word in the document. The chains are then broken in subchains if the distance between two consecutive words is longer than an empirically defined threshold. Subchains are then weighted based on two criteria: compactness and frequency. Higher weights are assigned when the subchain is short (compactness) and contains a high percentage of the words in the corresponding text span (frequency). LCseg also incorporates features related to multi-party discourse segmentation. These features include cue phrases, speaker change, silences, and overlapping speech. LCseg was evaluated in the ICSI meeting corpus (Janin et al., 2003) using the WindowDiff (WD) metric (Pevzner and Hearst, 2002), which is a penalty metric between 0 and 1 (the lower, the better). WD slides a window across the document and compares the number of hypothesized segments with the reference. The higher the discrepancy is between the number of segments in a window, the higher will be the penalty. LCseg obtained an average WD of 0.35, improving by 22.92% a baseline using the C99 algorithm (the segmentation approach we describe next).

The C99 algorithm (Choi, 2000) uses lexical similarity differently. The goal of the approach is to use a local context to refine a matrix built from the similarity of all possible pairs of utterances in the document. The refinement process ranks each entry of the matrix with the number of neighbors that have a lower similarity value. This provides a smoothing effect, which makes segments stand out (Figure 2.2). Then, a divisive clustering approach obtains the final segmentation. Choi evaluated C99 in a set of documents built by putting together sentences from different documents; in these artificially built documents, sentences from the same source correspond to segments. The results show precision scores between 88 and 91%, improving TextTiling between 32 and 39%. More recently, a new similarity metric, Content Vector Segmentation (CVS), based on the average value of the word embeddings of a segment, is used in the C99 algorithm (Alemi and Ginsparg, 2015). This representation is based on the average value of the word embeddings of a segment. When compared with the bag-of-words representation (also using the C99 algorithm) improvements were obtained in Choi's dataset.



(a) Raw utterance similarty matrix. (b) Smoothed utterance similarty matrix.

Figure 2.2: The figures above (Choi, 2000) show the smoothing effect of applying a neighbor rank-based transformation to a similarity matrix. Darker shades depict lower similarity values, and lighter ones high similarity values.

In another line of research, Wang et al. (2017) combined a learning to rank framework and a CNN neural network to learn a semantic coherence ranking function between text pairs. This function explores two partial ordering relations of coherence between text pairs that are expected to hold. The first one states that the coherence score of text pairs from different documents is lower than those from the same document. The second one states that the coherence score of text pairs from different paragraphs is lower than those from the same paragraph. Using the previous formulations, a learning to rank task is setup with a ranking function $\sigma(\mathbf{w} \cdot tp_i) \rightarrow y_i$, where tp_i denotes a text pair, σ is the sigmoid function, y_i is the semantic coherence real value, and $\mathbf{w}$ is a weight vector. The neural network uses two symmetric CNN models to learn the representations of tp_i text pairs based on word embeddings. Given a number of target segments T, the final segmentation is determined by the T utterances with the lowest semantic coherence scores. Despite being a promising approach, state-of-the-art results were not achieved in the dataset provided by Jeong and Titov (2010), which contains documents in the domains of news articles, biographies, lectures (English as a second language podcasts), and biology (class assignment reports).

Also following an approach using neural networks, is the SECTOR algorithm (Arnold et al., 2019), which uses a topic embedding trained based on utterance topic classification. Following the network architecture from Koshorek et al. (2018), two stacked LSTM (Hochreiter and Schmidhuber, 1997) layers are used to decode word embedding representation of utterances. The output of the LSTMs is fed to a ‘bottleneck’ topic embedding layer. The last layer of the network is a softmax activation layer that provides the predicted utterance topic label. To recover segmentation, a TextTiling approach is applied to the topic embedding layer. This is done by computing the cosine difference between consecutive topic embedding vectors and looking for minimum values based on a predefined threshold. To improve accuracy, dimensionality reduction using PCA and Gaussian smoothing are applied (Ziou and Tabbone, 1998). The evaluation was carried in a dataset of Wikipedia articles about disease and cities. The reference segmentation is based on existing section headers. The approach improves C99 by 0.11 in WD. It should be noted that this is a fully supervised segmentation approach where it is necessary to both train on domain specific data and provide the target number of topics. Thus, for each new domain, it necessary to collect segmented data for training. This problem is further aggravated by the amount data required to train LSTM-based architectures for segmentation. We argue that while these approaches provide valuable insights about how neural networks can learn to solve segmentation in a supervised fashion, they are have little practical value. The contradiction is that if you need to rely on existing markers to obtain enough segmentation training data then you do not need a segmentation algorithm.

The previous approaches suffer from the problem of only modeling local textual dependencies. In addition, relying on a cutoff threshold for segmentation is also disadvantageous, since documents can exhibit both sharp and attenuated topic transitions throughout the text. Malioutov and Barzilay (2006) and Kazantseva and Szpakowicz (2011) propose dealing with these problems by explicitly modeling long-distance relationships between utterances in a document. In this context, Malioutov and Barzilay (2006) developed MinCut, an algorithm that frames segmentation in a graph-partitioning task. This is done by abstracting text into a weighted undirected graph $G = \{V, E\}$, where V is the set of nodes corresponding to utterances and the edges E represent the connections between all utterances in the document. The weight of the edge between two utterance nodes u_1 and u_2 corresponds to their lexical similarity $S(u_1, u_2)$ (Equation 2.1). The edges model long-distance relationships since they connect all sentences to every other sentence in the document. Given this setup, MinCut optimizes the k-way normalized cut criterion:

$$Ncut_k(V) = \frac{cut(A_1, V - A_1)}{vol(A_1)} + ... + \frac{cut(A_k, V - A_k)}{vol(A_k)}, \tag{2.2}$$

where k is the target number of segments, A_k is a set of nodes, $cut(A, B) = \sum_{u \in A, v \in B} S(u, v)$, and $vol(A) = \sum_{u \in A, v \in V} S(u, v)$. The *cut* and *vol* functions ensure that two partitions are not only maximally different from each other but also that the intra-partition similarity is maximal. The optimization task is carried out by resorting to a dynamic programing algorithm. The approach was shown to be more effective than C99 in two test cases. One using a dataset with video lecture transcripts from Physics classes, and another with slide presentations about Artificial Intelligence. MinCut obtained WD averages of 0.34 and 0.41 in these domains, while C99 obtained 0.38 and 0.42, respectively. MinCut also segmented Choi's dataset but underperformed when compared to C99, with an 11% decrease. The explanation for this result is that long-distance relationships do not exist at all in Choi's dataset, because it was built artificially.

Affinity Propagation Segmentation (APS) (Kazantseva and Szpakowicz, 2011) also models long distances relationships but uses affinity propagation clustering (Frey and Dueck, 2007), an approach based on a factor graph framework. A factor graph is comprised of a node variables for each pair of utterances. The functions (factors) associated with each node assign a similarity value to a the node and ensure a linear segmentation of the document. The blueprint of the algorithm is to maximize a multi-variate function by approximating it through a sum of simpler functions. The sum-algorithm (Bishop, 2006), a message passing-based procedure, is then used to find a configuration of variables that maximizes the objective function. APS obtained state-of-the-art results in the previously mentioned Artificial Intelligence domain, with a WD average of 0.4, improving the MinCut approach by 4%. An evaluation of a dataset with fiction books was also carried out. In this scenario, APS obtained a 0.35 WD average, a 3% higher score than MinCut.

2.2.2 Probabilistic Approaches

Probabilistic approaches to segmentation follow a topic modeling approach, with a setup that closely relates to Latent Dirichlet Allocation (LDA) (Blei et al., 2003). In the LDA model, it is assumed that words in a document are generated from a set of topics. These topics correspond to Categorical distributions over the vocabulary, commonly referred to as language models. Another assumption is that documents have their own topic proportions, which models how likely it is for a topic to occur in a document. Topic proportions also correspond to Categorical distributions, but over the set of possible topics. Having topic proportions entails that each document is composed of a mixture of topics, that is, words in the same document are generated from different topics. A Bayesian setup follows by using parametrized Dirichlet priors on both language models and topic proportions. Formally, the LDA model is described by the plate diagram in Figure 2.3, which encodes the following generative process[1]:

1. For each topic $k \in \{1, ..., K\}$, draw word distribution $\phi_k \sim \text{Dirichlet}(\beta)$.
2. For each document $d \in \{1, ..., D\}$,
   (a) Draw topic proportions $\theta_d \sim \text{Dirichlet}(\alpha)$.
   (b) For each word position $i \in \{1, ..., N\}$ in d:
      i. Draw topic:
         $z_{d,i} \sim \text{Categorical}(\theta_d)$
      ii. Draw word:
         $w_{d,i} \sim \text{Categorical}(\phi_{z_{w_{d,i}}})$

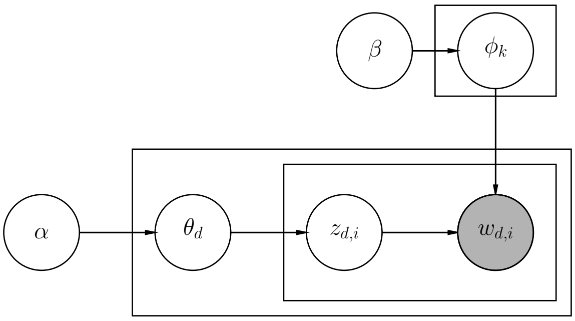


Figure 2.3: Plate diagram for the LDA model.

By examining the model setup, we can observe that to find useful topics sparsity is a necessary property. This means that topics have a set of words with a high likelihood, while the rest of the vocabulary rarely appears. For example, if we consider the topic of *sports*, it is likely that words such as ‘game’, ‘team’, ‘win’, or ‘ball’ frequently occur under this topic. A similar analogy can be made for topic proportions since we do not expect that a document discusses all topics, but instead focuses on just a few of them. This goes hand in

[1] We adopt the notational convention where variables denoted by Greek letters correspond to probability distributions. When the variables are uppercase they correspond to a set. Latin uppercase letters correspond denote size.

hand with the Dirichlet priors since its parameters can be set up to encourage a high likelihood of generating distributions with most of the probability mass concentrated in a few topics. Another reason for using Dirichlet priors is that they enjoy conjugacy properties when combined with the Categorical distribution, meaning that the resulting distribution is also Dirichlet. This is convenient since it simplifies the inference step in which we are determining the best parameter configuration of the unobserved variables[2] (non-shaded nodes in plate notation) that explains the observed data.

The original LDA model does not lend itself to segmentation, but its modeling assumptions still make sense in this task. The same sparsity characteristics we want for LDA topics can be used to implement lexical cohesion in a probabilistic setting. In the context of segmentation, this means we can achieve lexical cohesion if higher segmentation likelihoods have probability mass concentrated in a narrow subset of words. This can be done by constraining the inherent topics to the linear discourse structure. This is the core assumption of the probabilistic approaches we describe next, for both the single-document (Section 2.2.2.1) and multi-document (Section 2.2.2.2) cases.

2.2.2.1 Single-Document Segmentation

An example of a single-document segmentation approach using a topic modeling perspective is PLDA (Purver et al., 2006), where topic proportions are shared by sentences within the same segment. Words are generated by first drawing topics from segment topic proportions and then getting the actual word from the language model associated with that topic, which resembles LDA's generative process. Segmentation is determined through a binary topic shift variable associated with each sentence. The topic shift variables act as flipping a coin for each utterance in the document. If the result is heads, a new segment begins, and new segment topic proportions are drawn for this new segment. Topic shift variables are modeled with Bernoulli distributions, and a Beta prior is assumed. The beta prior variable allows encoding prior knowledge regarding the possible length of the segments. PLDA was evaluated in the ICSI meeting dataset. Comparable results to LCseg were obtained, with a 1% increase in performance. Later in this thesis (Section 4.2.2), we proposed an extension to PLDA for the multi-document segmentation case.

Structured Topic Model (STM) (Du et al., 2013) builds on the PLDA model by explicitly assuming that there is a topic structure between segments. Text passages in a segment still share the same topic distribution, but these are encouraged to be similar via a hierarchical prior. The intuition is that topic segment proportions are high level variations of document topic proportions. This captures how segment topic proportions

[2] Also commonly referred as latent or hidden variables in the literature.

relate to each other, instead of being independently generated, like in PLDA. In this context, segment topic proportions are generated from a Pitman-Yor Process (PYP) (Pitman and Yor, 1997). PYPs act like distribution generators, similarly to a Dirichlet, but allow to control how similar the distributions are to an input base distribution. In STM a document has its base distribution drawn from a Dirichlet, which is then fed to the PYP. The output of PYP is the final segment topic proportions. STM was evaluated in four different domains and benchmarked with C99, PLDA, APS, MinCut, and Bayesseg (an approach we describe later). The domains were the ICSI dataset, fiction books, election debates, and a clinical textbook. State-of-the-art results were obtained in all domains, except for the fiction domain, where Bayesseg performs best. The average WD results ranged from 0.38 and 0.26, obtaining improvements between 0.005 and 0.02.

Other works that also follow the PLDA research line include Nguyen et al. (2012), Riedl and Biemann (2012), and Jameel and Lam (2013). The SITS model (Nguyen et al., 2012) follows the PLDA approach but assumes that each text passage is associated with a speaker identity that is attached to the topic shift variable as supervising information. SITS further assumes speakers have different topic change probabilities and models them through a prior on topic shift variables. The model was evaluated in the ICSI and election debates datasets, but it does not improve STM. TopicTiling (Riedl and Biemann, 2012) uses LDA as a preprocessing step, where utterances are used as textual units instead of documents. The obtained topic assignments counts are then used as input for the TextTiling algorithm. The results on Choi's dataset show that TopicTiling could not achieve state-of-art results if the experimental conditions did not provide the target number of segments. Lastly, NTSeg (Jameel and Lam, 2013) focuses on breaking the typical topic modeling bag-of-words assumption by preserving word orderings of sentences. NTSeg was evaluated in the Physics dataset and the clinical textbook, but state-of-art results could also not be achieved.

The previous probabilistic approaches are mixed-membership models since words in the same segments can belong to different topics. Bayesseg (Eisenstein and Barzilay, 2008) takes a different modeling perspective on segmentation by casting it in a mixture model. This means all words belonging to the same segment are assumed to have been generated from a single word distribution language model (topic). A direct consequence of this approach is that a binary topic shift variable is not necessary since segmentation boundaries match precisely the places in the document where a topic change occurs. The topics continue to be assumed to have been generated from a Dirichlet prior. Given the previous setting, we can integrate out the language models. Therefore, the only remaining latent variables are topic assignments defining the segment structure of a document. This allows Bayesseg to perform inference using a maximum likelihood procedure. The procedure is based on a dynamic programing algorithm that efficiently explores the segmentation space and finds the latent variable configuration that maximizes the joint likelihood. This contrasts with the other approaches described thus far, which resort to Gibbs Sampling for inference. In the mixed-membership setting, there are two types of latent variables (topic shift and topic assignment variables), making an approach like the one in Bayesseg intractable. The advantage of Gibbs Sampling though is that it accesses the true posterior of the model, which leads to better parameter estimations. Bayesseg was originally evaluated in the ICSI dataset and a clinical textbook. WD averages of 0.31 and 0.35 were obtained, respectively. These results improved the LCseg baseline by 0.1 and 0.03. Later in this thesis (Section 4.2.1), we proposed an extension to Bayesseg for the multi-document segmentation case.

HierBayes (Eisenstein, 2009) is a follow-up work from Bayesseg, where segmentation is treated as a multi-scale lexical cohesion phenomenon. This means some words can frequently occur in a specific set of segments. In extreme cases, some words occur throughout the whole document, and others are segment specific. This setting lends itself to view segmentation as a hierarchical structure rather than a linear structure. To model a hierarchical structure, HierBayes organizes language models in a pyramid shape (Figure 2.4). The higher levels of the pyramid model words that span through larger parts of the document and lower-levels explain local sets of words. Inference is carried out in a two-step iterative process. First, a dynamic programming procedure finds the hierarchical segmentation that yields the highest likelihood, similar to Bayesseg. The main difference is that the likelihood is computed in a variational inference setting where each word is assigned a variational parameter representing the hierarchy level latent variables. The second step updates the variational parameters according to the current state of the segmentation. This segmentation/update loop continues until convergence. In the reported experiments HierBayes outperformed the best baseline, Bayesseg, with a 5.3% WD improvement when segmenting a clinical textbook.

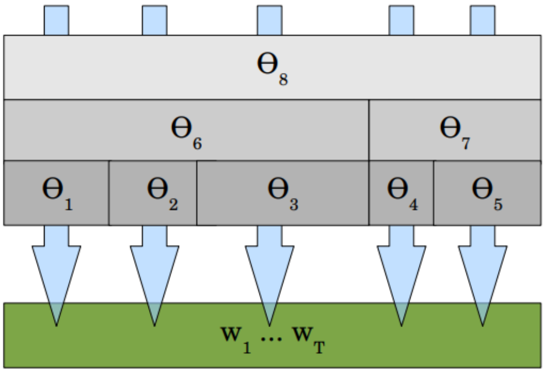


Figure 2.4: HierBayes hierarchical structure example for a document with length T. The structure describes the document with two main segments (θ_6 and θ_7), which can be further be broken down into three (θ_1, θ_2, and θ_3) and two (θ_4 and θ_5) more fine grained segments, respectively.

2.2.2.2 Multi-Document Segmentation

Thus far, the described probabilistic segmentation research work does not afford multi-document segmentation nor topic identification, the main goals of this thesis. To the best of our knowledge, the only work that addresses these issues is the MultiSeg model (Jeong and Titov, 2010). The model can be seen as a hybrid between a mixture and mixed membership model. At the document level, a topic proportions variable explains how likely is a topic to occur in the document. The model then constrains segments to have a single topic. MultiSeg assumes that documents are generated by two different types of topics: local and global. Local topics are associated with a single segment specific to a document. Global topics can be shared across all documents, allowing multi-document segmentation and topic identification to be achieved. Both types of topics are generated from topic proportion variables obtained from a Dirichlet Process (DP) prior. The full generative process of MultiSeg is described as follows:

1. Draw global topic proportion $\alpha \sim \text{DP}(\alpha_0)$
2. For each topic $k \in K$:
    (a) Draw global language model $\phi_k \sim \text{Dirichlet}(\phi_0)$
3. For each document $d \in D$:
    (a) Draw local topic proportion $\beta^d \sim \text{DP}(\beta_0^d)$
    (b) For each topic $j \in K$:
        i. Draw local language model $\psi_j^d \sim \text{Dirichlet}(\psi_0^d)$
    (c) Draw $\eta^d \sim \text{Beta}(\eta_0^d)$
    (d) For each utterance $u \in d$:
        i. Chose topic type $z_u^d.t \sim \text{Bernoulli}(\eta^d)$
        ii. If $z_u^d.t = SHARED$ then chose topic $z_u^d.l \sim \alpha$;
        generate words $x_u^d \sim \text{Categorical}(\phi_{z_u^d.l})$
        iii. Otherwise, chose topic $z_u^d \sim \beta^d$;
        generate words $x_u^d \sim \text{Categorical}(\psi^d_{z_u^d.l})$

From the description of the generative process, we can see that MultiSeg manages the use of global and local topics with a Bernoulli distributed variable. Depending on the $z_u^d.t$ topic type drawn from the Bernoulli, the corresponding global or local topic proportions (α or β^d) will then generate a topic for the utterance. It should be noted that the only goal of the DP is to determine the number of existing local and global topics. This contrasts with the use of PYP in STM, where it is possible to both determine the number of topics as well as encourage topic proportion similarity. The difference is that MultiSeg directly uses the draws from the DP as topic proportions while STM uses another Dirichlet prior to obtain hyperparameters that are used as input of PYP. The evaluation was carried out in four different domains: biology student reports, news articles, biographies, and English as a Second Language (ESL) podcasts. The evaluation showed WD improvements ranging from 1.8% to 17.8% compared to a Bayesseg baseline. The results in topic identification evaluation showed F_1 improvements between 18.9 and 30% over a pipeline approach using k-means clustering. These results provide evidence that the tasks should be modeled jointly.

2.2.3 Conclusions

After describing the research context, we conclude that the tasks of multi-document segmentation and topic identification are not very well studied research lines in the document segmentation area. Despite these problems lending themselves to a probabilistic framework, existing approaches, with the exception of MultiSeg, cannot be used to model segmentation in this perspective. In mixed memberships models, topics are shared across segments, but each of them is comprised of several topics generated from their individual segment topic proportions, and, thus, no topic similarity can be directly extracted. Mixture models in a single-document segmentation context assume that each segment is generated by an individual topic, which does not allow topic identification. Despite MultiSeg being a multi-document model, it assumes that all segments are independent of each other. Under these conditions, it is not possible to study our vocabulary relationship across segments in different documents hypothesis. This motivates us to propose a model that encodes such assumptions, and, consequently, allows to obtain answers for our research questions.

3 Dataset Collection

Motivated by the lack of resources to evaluate multi-document segmentation and topic identification, especially in the context of documents that develop an overarching topic, we carried out a data collection task. We start this chapter by describing existing datasets and discuss their limitations (Section 3.1). Then, we describe the methodology used to gather a suitable dataset to evaluate the tasks (Section 3.2). Finally, in Sections 3.3 and 3.4, we perform an inter-annotator agreement study to evaluate to what degree human judges agree on segmentation and topic identification.

3.1 Available Datasets

The available datasets for document segmentation focus on the single-document case, which does not translate well for the multi-document segmentation. This is because multi-document segmentation models assume that the same topics are discussed in several different documents. The datasets for single-document segmentation do not have this property. Although they might have an overarching domain, the topics discussed in the segments are inherently different. For example, Malioutov and Barzilay (2006) provide a dataset in the Physics domain. Despite all documents being in the same overarching domain, the subjects are different. For example, the topics described in the segments from the 'Hooke's Law' subject are not the same as in the 'Work and Energy' subject (the following lesson). Therefore, the segments describe different topics as well. What is needed for evaluating multi-document segmentation are documents describing similar topic segments, which we refer as related documents. It should be noted that this neither implies that all documents have all topics, nor that they are all described at the same level of detail.

The discussion below focuses on the datasets from the following works: Choi (2000), Janin et al. (2003), Kazantseva and Szpakowicz (2011), Eisenstein (2009), Malioutov and Barzilay (2006), Ward et al. (2013), Joty et al. (2013), and Jeong and Titov (2010).

A dataset typically used to evaluate segmentation is the one developed by Choi (2000). This is an artificial dataset comprised of the first n sentences from different documents from the Brown corpus (Francis and Kucera, 1979). This corpus contains documents from 500 sources categorized by domain (politics, sports, *etc.*). In total, 700 artificial documents were generated, each with ten segments, and using a value of n between three and eleven. The problem is that topic boundaries correspond to abrupt changes in the vocabulary, which does not capture the topic development aspect where word usage changes smoothly.

The ICSI dataset (Janin et al., 2003) consists of multi-party meetings from various research groups, such as the natural language processing and internet architectures groups. In total, the dataset contains 75 meeting transcripts, of which 25 were manually segmented. The average number of segments per documents is five. The problem with the dataset is that the meetings mostly discuss ongoing research projects for which it is hard to find related documents. For instance, the meetings of the natural language processing group concern the building of the ICSI dataset itself. Later, Hsueh et al. (2006) extended the annotations in this dataset to further divide the existing segments into finer-grained topics, but the problem remains.

Kazantseva and Szpakowicz (2011) provided a dataset with 22 chapters from a XIX century novel, *The Moonstone* by Wilkie Collins. The chapters have an average length of 53.85 paragraphs, with an average of 5.8 segments each, and were annotated by groups between four and six annotators. Annotators were instructed to segment the chapters into episodes, in order to create an outline for the chapter. Given the inherent uniqueness of a book novel, it not possible to find segments describing the same topic.

In an educational setting, Eisenstein (2009) used a medical textbook (Walker, Dallas, and Willis 1990) to evaluate topic segmentation. The goal was to split each chapter in their sections. This dataset contains 227 chapters, with 1136 sections (an average of 5 per chapter). Each chapter includes an average of 140 sentences, giving an average of 28 sentences per segment. No related documents are available in this corpus.

Also in the educational setting, Malioutov and Barzilay (2006) provided two lecture datasets in the domains of Artificial Intelligence (22 lectures with 5.9 segments in average) and Physics (33 lectures with 12.3 segments in average). Considering the former, topic boundaries in each video transcript were based on the slide changes occurring in the video. Therefore, these boundaries may not correspond to topics. In the latter case, documents were manually segmented by four annotators. In both cases, each lecture has its own domain, thus, multi-document topic segmentation models cannot be evaluated using this data.

The Similar Segments in Social Speech (4S) task Ward et al. (2013) provided a corpus with topically related segments. The task is defined as receiving an audio segment of interest and returning an ordered list of jump-in points for regions similar to it. By doing so, whenever a relevant piece of information is found, other related information can also be accessed efficiently. For this purpose, a corpus of dialogs among university students was created. The dataset has 26 dialogs with 309 minutes in total (1697 segments). Dialogs were annotated by four annotators, though the same dialog was not annotated by more than one annotator. Topic identification of the segments is also available. What makes the 4S corpus unsuitable for our tasks is that there is no general subject in the individual documents (for example, the same document can talk about internships, family, movies, *etc.*). This is the same issue described in Choi's dataset.

Joty et al. (2013) studied topic segmentation and labeling in asynchronous conversations, that is, conversations that do not require right-now attention. The provided dataset contains 40 email threads from the World Wide Web Consortium with an average number of 2.5 segments per thread. Although each email thread has repeated topic discussions, several unrelated topics also occur in a single thread. This has the same pitfalls identified in both 4S and Choi's datasets: segments can be easily distinguished, which do not allow an accurate evaluation. In addition, the dataset contains 20 conversations, with an average of 10.8 segments each, from the technology-related news website Slashdot. In this domain, a document corresponds to the sequence of comments made by users regarding a news article. This scope is much different from the one we want to target in this thesis: how documents describing in detail some subject are structured.

Considering multi-document model evaluation, the available datasets have limitations. Multiseg (Jeong and Titov, 2010) was evaluated in four different domains: News, Biography, Report, and Podcast. To create the News domain, document clusters were collected from the science and technology section in `news.google.com`. The data for the Biography domain comes from four websites: `en.wikipedia.org`, `simple.wikipedia.org`, `biography.com`, and `notablebiographies.com`. The Report domain consists of reports describing a plant growth lab, an assignment for a biology class (Sun et al., 2007). The Podcast domain, corresponds to the English as a second language podcast (Noh et al., 2010). Each episode consists of two documents: a story and an a podcast lecture with a teacher and a student discussing the meaning and usage of English expressions appearing in the story. The goal was to divide the lecture transcript into discourse units. Topic identification of the segments is available for all domains. Table 3.1 provides statistics for each individual domain. Despite the high number of documents, it is important to notice that what we care about the most is the number of documents per subject (*RelatedDocs* column). In this perspective, each subject has, in fact, few documents. The only exception is the *Report* domain, but the number of segments per document is low.

Domain	**#Subjects**	**#Documents**	**RelatedDocs ($\overline{x}$)**	**Segments ($\overline{x}$)**	**#Topics**
News	50	184	3.7	3.0	220
Biography	30	120	4.0	8.1	405
Report	1	160	80.0	2.4	2
Podcast	200	400	2.0	18.2	3819

Table 3.1: Dataset statistics from Jeong and Titov (2010). The symbol $\overline{x}$ indicates an average.

3.2 Learning Materials Dataset

Given the limitations of available datasets, we carried out a data collection task to obtain a dataset where topic development could be observed across related documents. During this process, we target a dataset representing the needs of a real-world user. For this purpose, we adopt the scenario of a student, who needs to read about different subjects in some domain. Therefore, he retrieves several documents covering those subjects from the web and briefly decides which ones should be part of the collection. We implement this scenario by gathering related documents from different modalities from the web, using a web scraping approach. The approach consists in extracting keywords (Bougouin et al., 2013) from seed documents and submitting them to the following search engines: Google, Bing, Yahoo, and DuckDuckGo. Working from the top retrieved results down, documents are added to the dataset by balancing the number of documents from each modality as much as possible. All unrelated documents were discarded. The main content of the documents was extracted to a text file using available software[1,2,3]. In what concerns videos, we extracted the corresponding subtitles.

The previous process was first carried out to collect a small scale dataset in the subject of Adelson-Velsky and Landis (AVL) trees (Adelson-Velsky and Landis, 1962), a topic often found in Computer Science curricula. Instead of using a seed document, the keywords 'AVL trees' were submitted to the search engines. A summary of the statistics of the dataset is in Table 3.2, in which:

- #Docs is the number of documents.
- |Doc| is the average length of a document in sentences.
- #Seg is the average number of segments per document.
- |Seg| is the average length of a segment in sentences.
- #Words is the average number of words per segment.
- #Topics is the number of topics.
- |Vocab| is the length of the vocabulary.

All of the documents refer to the topic of AVL trees except for one HTML document, a Wikipedia article, which specifically addresses tree rotations (an essential operation in AVL trees). The corpus was

[1] https://github.com/codelucas/newspaper
[2] https://tika.apache.org/
[3] https://rg3.github.io/youtube-dl/

segmented into topically cohesive segments by the author of the thesis, who has taught AVL trees in an Algorithms and Data Structures course. General rules of thumb provided in other studies (Galley et al., 2003; Kazantseva and Szpakowicz, 2011; Malioutov and Barzilay, 2006) were followed, namely: a) Segments should be cohesive and self-contained; b) Segments should contribute to the understanding of the content organization of the document. In total, 86 segmentation boundaries and 17 topics were annotated, from a corpus containing 3181 sentences.

	#Docs	**|Doc|**	**#Seg**	**|Seg|**	**#Words**	**#Topics**	**|Vocab|**
PPT	5	202.2±69.2	7±1.2	33.4±1.4	1402.0±683.5	6	776
HTML	2	68.5±0.7	7±1.4	10.6±5.8	1195.5±221.3	5	536
Video	3	675.6±77.4	11±5.5	64.2±57.9	6396.3±307.1	8	1265

Table 3.2: AVL trees dataset statistics.

Moving to a larger scale data collection task, we wanted to have a variety of subjects to test the robustness of segmentation algorithms. Another aspect taken into consideration is to use seed documents from a dataset that is already familiar, and that has been validated by the segmentation research community. In this context, we chose seven documents, from the Physics lectures dataset (Malioutov and Barzilay, 2006), to be used as seed documents. The following documents were used:

- L02 - Introduction to Kinematics
- L03 - Vectors
- L06 - Newton's Laws
- L08 - Frictional Forces
- L10 - Hooke's Law
- L11 - Work and Energy
- L20 - Angular Momentum

After gathering the documents using the previous web scraping approach, we followed the same manual segmentation task performed for the AVL trees. Statistics regarding the Physics lecture dataset can be found in Tables 3.3 and 3.4 (dataset by subject and by modality, respectively).

Given the size of the collected dataset, we achieve the goal of affording a more thorough evaluation of multi-document segmentation and topic identification, overcoming, to some extent, the problems identified

Subject	**#Docs**	**|Doc|**	**#Seg**	**|Seg|**	**#Words**	**#Topics**	**|Vocab|**
$L02$	19	150.84±119.2	4.8 ±2.6	31.5 ±27.3	508.5 ±454.9	27	3210
$L03$	21	128.38 ±97.3	5.3 ±2.2	24.3 ±23.1	358.6 ±387.7	12	3170
$L06$	20	176.6 ±121.9	6.1 ±2.9	28.9 ±30.0	421.4 ±468.2	17	4142
$L08$	19	133.6 ±115.8	4.7 ±3.2	28.2 ±32.4	455.2 ±562.7	21	3154
$L10$	20	108.3 ±97.80	4.3 ±3.5	25.2 ±30.2	411.9 ±467.1	17	2951
$L11$	21	164.8 ±109.6	6.7 ±4.4	24.7 ±23.1	378.3 ±398.5	15	4153
$L20$	21	114.5 ±100.8	4.7 ±2.6	24.5 ±20.5	477.5 ±481.0	26	3892

Table 3.3: Physics dataset by subject.

	#Docs	**|Doc|**	**#Seg**	**|Seg|**	**#Words**	**#Topics**	**|Vocab|**
HTML	60	96.4 ±81.3	2.5 ±2.3	20.5 ±17.5	367.5 ±330.2	85	6614
PPT	27	176.4 ±80.9	8.1 ±3.9	21.8 ±17.8	197.9 ±163.5	62	4401
PDF	15	169.1 ±110.4	7.5 ±1.9	22.65 ±10.6	490.3 ±251.9	32	5349
Video	39	168.7 ±142.8	3.2 ±2.6	52.23 ±42.7	888.40 ±673	55	4176

Table 3.4: Physics dataset by modality.

in existing datasets. Contrary to the datasets surveyed in Section 3.1 (Table 3.5), our learning materials dataset was tailored for testing multi-document segmentation models in collections of related documents from different modalities. In total, we gathered 151 documents in 8 different subjects (7 in the Physics' domain plus the AVL Trees domain). In the Physics domain, each subject has around 20 documents, with an average number of 5 segments per document in a total of 739 segments, which makes it larger than the other datasets, apart from Choi's, Eisenstein and Minwoo datasets. However, Choi's dataset is artificially generated; the 227 documents from Eisenstein are the chapters of a single book; Minwoo's dataset is composed of documents on four different domains that either do not have a high number of related documents or the number of segments is low (the Report dataset).

Dataset	**Content**	**#Documents**	**#Segments**
Choi	News	700	7000
ICSI	Meeting transcripts	25	188
The Moonstone	Book Novel	22	128
Eisenstein	Medical textbook	227	1136
Malioutov	Lectures	55	536
4S	Student dialogs	26	1697
Shafiq	Emails and News	60	316
Minwoo*	Varied	864	9188

Table 3.5: Existing datasets with document segmentations. The dataset marked with '*' can be used for multi-document segmentation evaluation.

3.3 Human Segmentation Agreement Study

In order to study how different human judges segment our dataset, three more annotators were asked to annotate 16 documents (4 for each different modality) from the 'Introduction to Kinematics' and 'Hooke's Laws' subjects. For each subject, 8 documents were annotated. One of the judges annotated 16 documents. The remaining two annotated 8 documents each. Two of the new annotators also have a background in Computer Science, and the other has a Mechanical Engineering background. The annotators were given a deadline of two weeks to complete the annotation task. None of the annotators were monetarily compensated. Although the annotation was carried out in the text file version of the documents, a link to the original format (PPT, video, *etc.*) was provided. No particular order of the document was imposed to the annotators and they were free to revise annotations before committing to a final segmentation.

Again, the guidelines closely followed previous similar studies (Galley et al., 2003; Malioutov and Barzilay, 2006; Kazantseva and Szpakowicz, 2011), and explained the concept of a topic segment in a non-technical way[4]. Examples and counter-examples of document segmentations were provided. In addition, and similarly to Malioutov and Barzilay (2006), we asked annotators to give a short description of each segment. The idea is that if it is hard to come up with a description or if it is similar to the previous segment, then, probably, the boundary should not exist. No expected number of segments was provided. Finally, we asked annotators to indicate if they were certain (or not) about the boundary annotation. This procedure has not been done in previous studies, but it incentivizes annotators to indicate boundaries in case of uncertainty. We consider that uncertainty in the annotation is related to annotators having doubts if their segmentation matches the target level of granularity. By investigating the role of uncertainty in the annotation process, we can better determine if annotators are converging to the same segmentation.

3.3.1 Inter-Annotator Agreement Metrics

Previous agreement studies (Janin et al., 2003; Malioutov and Barzilay, 2006) have used the standard segmentation metrics to evaluate inter-annotator agreement, namely P_k (Beeferman et al., 1999) and WD (WindowDiff) (Pevzner and Hearst, 2002). These measures are preferable to the precision and recall used in classification. The problem of evaluating segmentation with classification metrics is the required strict match between hypothesis and reference. For example, precision and recall would indicate that a hypothesis with boundaries close to the appropriate location is worst than a 'degenerate' segmentation that places a boundary in all possible locations. A relaxation of the precision and recall that allows the boundaries

[4] A copy of the instructions can be found in Appendix C.

to be placed within some constant-sized window from the reference does not solve the problem. In this case, it would not be possible to distinguish a precisely accurate segmentation from another that always puts boundaries close to the reference. Agreement coefficients used in other annotation tasks, such as Cronbach's α (Cronbach, 1951), Scott's π (Scott, 1955), Cohen's k (Cohen, 1960), Fleiss k (Shrout and Fleiss, 1979), and Krippendorff's α (Krippendorff, 2004), fall under the same pitfalls as the classification metrics.

Given the previous context, the P_k metric was developed by Beeferman et al. (1999). The intuition for this metric is that a segmentation is better than another when it has a higher probability of correctly distinguishing whether two words belong to the same segment or not. Therefore, P_k corresponds to a probability, and, thus, lower values are preferable. P_k is derived from the following, more general, formulation:

$$P_D(ref, hyp) = \sum_{1 \leq i \leq j \leq n} D(i,j)\Big(\delta_{ref}(i,j) \bar{\oplus} \delta_{hyp}(i,j)\Big), \tag{3.1}$$

where ref is the reference segmentation of a corpus of n words, hyp is output segmentation of an algorithm, and δ is the indicator function, which returns 1 if the words i and j belong to the same segment and zero otherwise. The operator $\bar{\oplus}$ is the `XNOR` function ('both' or 'neither'), and the D function is a distance probability distribution over all possible distances between two randomly chosen words chosen. If D is a uniform distribution, then the metric is too forgiving, since the majority of the distances will be large, and, for these cases, even naive segmenters will perform accurately. It has been shown that using $P_D = P_k$ (Beeferman et al., 1999) (meaning that all probability mass is concentrated in a single fixed distance, k), yields a good evaluation metric for the segmentation task. In practice, this corresponds to defining a window with size k and using it to sweep the corpus while checking if words are correctly classified. By using this scheme, the notion of 'close to correct boundary' is captured in a principled way by smoothly penalizing segmenters that place boundaries that are not quite right, and by scaling with the segmenter's degradation.

Although P_k is a better segmentation evaluation metric than precision and recall, it still has drawbacks. Namely, it penalizes false negatives more heavily than false positives, over-penalizes 'near-misses', and is affected by the segment size distribution (Pevzner and Hearst, 2002). In dealing with these problems, a modification of the metric, WD, was proposed by Pevzner and Hearst (2002). The difference is that, in each window, the penalty is difference between the number of boundaries in ref and in hyp, instead of the binary assessment of whether the words are in the same segment or not. WD is formalized as follows:

$$\mathrm{WD}(ref, hyp) = \frac{1}{N-k} \sum_{i=1}^{N-k} |ref - hyp| \neq 0, \tag{3.2}$$

where N is the document length and k the window size. WD is a penalty score from 0 (best value) to 1.

Despite improving the P_k metric, WD introduces the problem of biasing better results towards segmentations with fewer segments (Fournier, 2013). In dealing with this issue, Fournier (2013) proposed the Boundary Edit Distance (BED) metric. BED models segmentation differences using additions/deletions (full misses), and transpositions (near misses). A transposition occurs if the annotated boundaries are off by at most n sentences. By explicitly defining what constitutes a full miss and near-miss situations, it is possible to operate at the individual utterance level (potential boundary). This avoids the problem of window-based metrics where we are looking at two windows without any context of the segments where they belong to. For example, we might have two windows without any boundary and give full credit when the underlying segments are actually quite different. This is the main cause of the bias problem in WD. The BED metric is a score between 0 and 1 (the best value) and is assigned depending on the number of boundary matches and edit operations needed to make the segmentations equal. The BED metric is defined as follows:

$$\text{BED}(ref, hyp, n_t) = 1 - \frac{|AD|+T_w}{|AD|+|T|+|B_M|}, \tag{3.3}$$

where n_t is the maximum distance that boundaries may span to be considered transpositions, AD is the set of addition/deletions, T the set of transpositions, T_w is a transposition penalty weight, and B_M the set of matching boundaries. T_w models the severity of a transposition my making the penalty harsher for larger n. In practice, this is done by dividing the distance between the boundaries by n_t and adding a constant value.

To use BED for annotator agreement, Fournier (2013) integrates it in the k_{Fleiss} coefficient (the multi-annotator version of π (Shrout and Fleiss, 1979)). This is possible because BED is symmetric, unlike P_k or WD. The metrics are combined by using the pairwise mean BED in k_{Fleiss}. This coefficient is based on the observed agreement (A_{obs}), corrected for chance by the expected agreement (A_{exp}). When there is no agreement k_{Fleiss} returns 0, and for complete agreement it returns 1. k_{Fleiss} is calculated as follows:

$$k_{Fleiss} = \frac{A_{obs} - A_{exp}}{1 - A_{exp}} \tag{3.4}$$

Adapting A_{obs} and A_{exp} to use the BED metric, k_{BED}, results in:

$$\begin{aligned} A^B_{obs} &= \frac{1}{\binom{C}{2}} \sum_{m=1}^{C-1} \sum_{n=m+1}^{C} \frac{\sum_{d\in D}|d|\text{BED}(s_{dm}, s_{dn})}{\sum_{d\in D}|s_d|-1} \\ A^B_{exp} &= \frac{\sum_{c\in C}\sum_{i\in I}|s_{dc}|}{C\sum_{d\in D}|d|-1} \\ k_{BED} &= \frac{A^B_{obs} - A^B_{exp}}{1 - A_{exp}}, \end{aligned} \tag{3.5}$$

where c is an annotator from the set of annotators C, d is a document from a dataset D, $|d|$ is the size of d (in utterances), s_{dc} is the segmentation of d provided by annotator c, and $|s_{dc}|$ is the number of segments in d annotated by c. In the carried out agreement study, we report WD, k_{BED}, and k_{Fleiss} to compare our results with previous agreement studies.

3.3.2 Inter-Annotator Agreement Results

The number of obtained certain/uncertain boundaries for each individual annotator, for each subject, is presented in Tables 3.6 and 3.7. The second annotator, A2, was the one responsible for annotating the whole collection. In total, 240 segment boundaries were annotated, from which 47 were marked as uncertain (19.6% of the total).

	#Certain	#Uncertain
A_1	23	23
A_2	36	8
A_3	37	2

Table 3.6: Number of certain and uncertain annotations for $L02$ documents.

	#Certain	#Uncertain
A_2	35	5
A_3	37	5
A_4	25	4

Table 3.7: Number of certain and uncertain annotations for $L10$ documents.

Considering that we have certain and uncertain boundaries, and in order to bring 'crowd wisdom' to the annotation process, uncertain boundaries marked by a single annotator were discarded. The intuition was the following: if only one annotator marks a boundary as uncertain, then it is plausible to assume that the boundary may not exist. On the other hand, if multiple annotators mark a boundary as uncertain, it is plausible to assume that it exists. Tables 3.8 and 3.9 show the inter-annotator agreement results using the previously described metrics, for each subject. For completeness, we also report the agreement results using the strict metric k_{Fleiss}.

	k_{BED}	WD	k_{Fleiss}
HTML	0.81±0.20	0.20 ±0.1	0.89 ±0.11
PPT	0.58±0.02	0.17±0.1	0.70±0.04
PDF	0.40±0.01	0.20 ±0.1	0.55 ±0.02
Video	0.67±0.30	0.36±0.1	0.75±0.25

Table 3.8: Inter-annotator agreement in $L02$ documents.

Despite the differences in the results between k_{BED} and k_{Fleiss}, ranging between 0.08 and 0.12, both metrics provide a similar view of the agreement obtained in the annotation tasks. What stands out from these results is that k_{Fleiss} yield higher agreement than k_{BED} when previous reports showed an opposite

	k_{BED}	WD	k_{Fleiss}
HTML	0.91±0.09	0.11±0.00	0.95±0.05
PPT	0.55±0.15	0.26±0.11	0.67±0.15
PDF	0.61±0.12	0.18±0.20	0.73±0.09
Video	0.43±0.29	0.21±0.06	0.53±0.29

Table 3.9: Inter-annotator agreement in $L10$ documents.

behavior (Fournier, 2013). This is related to the fact that we removed uncertain boundaries marked by a single annotator. Regarding the WD results, they do not entirely agree on which is the hardest and easiest modality to annotate when compared with the other agreement metrics. For example, in $L02$ the k_{BED} results indicate that HTML documents obtained higher agreement and PDF the lowest. For WD, the highest agreement is obtained in PPTs and the lowest in Video transcripts documents, although the difference between PPT and HTML is only 3%. Less ranking discrepancies were observed in the $L10$ case. Only the ranking of PPT and PDF is not consistent. On a manual qualitative analysis of the annotations, we noted that k_{BED} better translates the agreement differences across media sources. This is in line with Fournier (2013), where it is argued that WD overinflates its agreement results for sparse segmentations, which is indeed the case for documents that show inconsistencies across evaluation metrics.

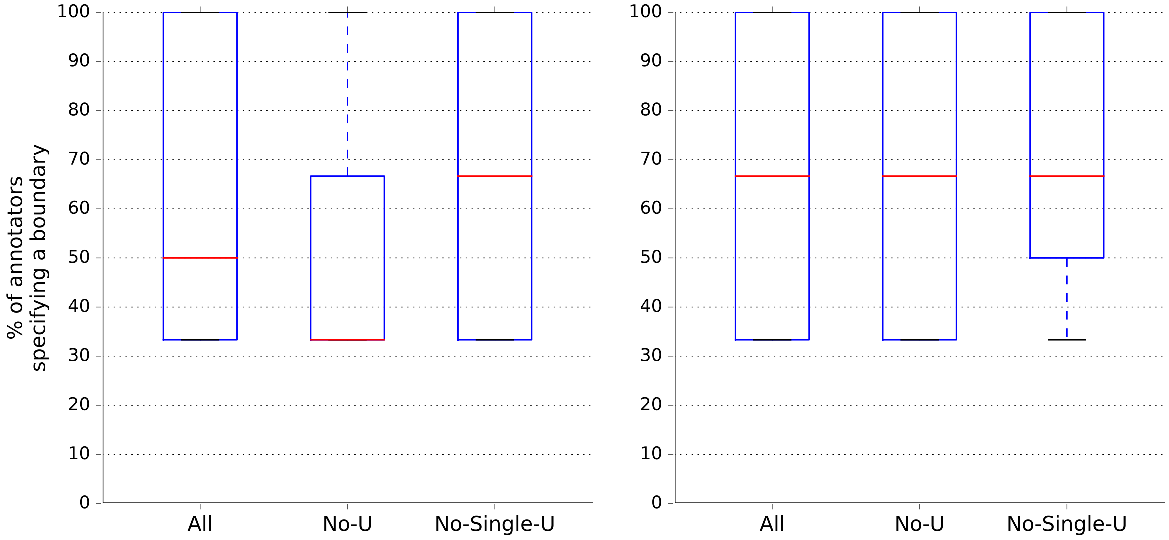
k_{BED} results are higher than the values reported in other datasets. In total, we obtained a 0.65 average agreement, whereas Fournier (2013) and Kazantseva and Szpakowicz (2011) obtained 0.44 and 0.30, respectively. A similar scenario is found when comparing WD results. We obtained a 0.21 average WD, which is better than the 0.35 in Janin et al. (2003) and 0.34 in Malioutov and Barzilay (2006). Some of the differences between our annotation task and the previous ones, which might explain the different results, are the nature of the target documents and how segments are annotated. Our dataset is comprised of learning material documents which have a less ambiguous topic structure than the book chapters of the romance in Kazantseva and Szpakowicz (2011). Our annotation task is also simpler than the one in Malioutov and Barzilay (2006), which added extra cognitive load by requiring the identification of major and minor topics.

3.3.3 Dealing with Uncertainty

Previous results were calculated considering that uncertain boundaries marked by a single annotator are discarded (from now on the No Uncertainty Singleton (No-Single-U) scenario). In order to study the impact of uncertain boundaries, we conducted another study, in which two other scenarios were considered:

- All Annotations (All): all boundaries (certain or uncertain) are considered.
- No Uncertainty (No-U): all uncertain annotations are discarded.

Figure 3.1 shows what percentage of annotators included a particular boundary in their segmentation for each version of the annotations dataset and each individual subject.



(a) Annotations of the $L02$ documents.

(b) Annotations of the $L10$ documents.

Figure 3.1: Box plots for the percentage of annotators assigning a topic boundary.

For the $L02$ case (Figure 3.1a), we can see that the No-U scenario is the one with more disagreement among annotators, since most boundaries were annotated by 33% and 67% of the annotators, and only in a few cases there is complete agreement. This result contrasts with the All and No-Single-U annotations, where a higher percentage exists, as most boundaries were annotated by two annotators. Moreover, a significant portion of the distribution has 100% agreement. The difference between All and No-Single-U is the median value. In No-Single-U the median is 67%, and, thus, there are more cases where at least two annotators specified a boundary.

The differences between the annotation datasets for $L10$ are not as prominent as before (Figure 3.1b). In all scenarios, a median value of 67% was obtained. The difference is that the No-Single-U annotations do not have as many lower percentage values as the remaining. These different results in $L02$ and $L10$ suggest that some subjects might be harder to segment than others. This was hinted before when looking at the number of uncertain boundaries used in $L02$ and $L10$ (Tables 3.6 and 3.7). For the latter case, a much more varied number on uncertain boundaries was used, even though two of the annotators remained the same across subjects. This argues in favor of the proposed annotation scheme which allows a post-analysis of the annotations that better translates the actual agreement between annotators.

3.3.4 Qualitative Analysis

To better understand the disagreement patterns, Figure 3.2 shows a plot with the annotated boundaries in a PDF document with a low agreement ($k_{BED} = 0.4$) from the $L02$ subject. Despite the low agreement, it is possible to observe that there is a set of boundaries for which all the three annotators agree (the vertical lines with three different colors). This indicates that some topical shifts are more prominent than others.

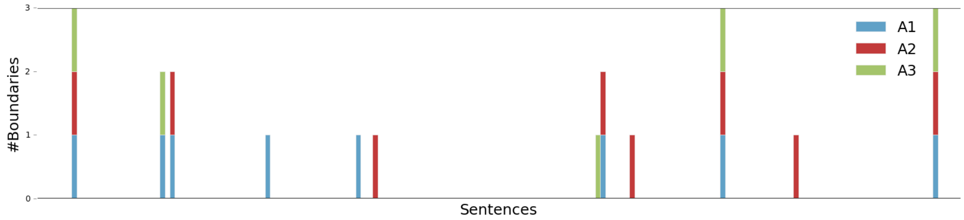


Figure 3.2: PDF document annotations in a document from the $L02$ subject.

Figure 3.2 also illustrates "near miss" situations, in which the boundaries chosen by the annotators are close (the vertical lines with different colors are near to one another). This is due to the difficulty in perceiving if a text span is concluding the current segment or introducing the following one. This usually happens when a text span compares and contrast concepts from the previous and next segments.

From Figure 3.2 it is also possible to see that different levels of granularity still exist in some annotations. Annotator `A1` provided a finer level of granularity than the remaining ones. For example, he considered two different segments of 'Average Velocity' and 'Instantaneous Velocity', whereas the remaining annotators considered a single segment ('Velocity').

The granularity issue observed in our study has also been reported in previous works (Passonneau and Litman, 1997; Kazantseva and Szpakowicz, 2011; Fournier, 2013), and pointed out as the main reason for the low inter-annotator agreement when compared to other annotation tasks. These annotation differences stem from the different views in what the segmentation granularity should be, and not necessarily with the difficulty in identifying topics in a document. Also, this is not the general case for all topic segments: as discussed before, there is a considerable number of prominent topic shifts for which the annotators agree.

To give an intuition of why annotators disagree, Figure 3.3 shows the textual representation of the beginning of the PDF document from Figure 3.2. This is an example where the three annotators had different segmentation perceptions regarding the concept of *average speed*. Annotator `A1` marked a dedicated segment for this concept, showing a finer level of granularity, whereas `A2` and `A3` agglomerated it with the next and previous segments, respectively. This demonstrates the ambiguity that can be encountered during the segmentation annotation task. Even when not considering *average speed* in a single segment, it is debatable if whether it is more related to the *distance* segment (the previous) or to the *average velocity* segment (the next). Despite these hard to judge cases, Figure 3.3 also shows examples where prominent topic shifts occur. For example, all annotators marked a first segment and described it as *introduction*. Another interesting observation is that all annotators considered *position* and *displacement/distance* to be part of the same segment. This indicates that segments are constructs that can contain several concepts that together form a cohesive piece of information.

3.4 Human Topic Identification Agreement Study

The inter-annotator agreement for topic identification was carried out as a clustering task. Annotators were instructed to group segments if they shared the same topic. Since it is possible that a single segment contains multiple topics we allow annotators to have a segment in multiple clusters. The instructions also specified that an exact semantic match in the content of two segments is not necessary in order to consider that they share a topic. Therefore, some segments can develop a topic more than others and still belong to the same cluster. Following the previous segmentation annotation task, examples and counter-examples of these cases were provided in the instructions. Annotators provide topic identification judgments in the same document segmentations. A setup where annotators use their own document segmentations would be possible if we consider topic identification judgments at the utterance level. The problem is that differences in the segmentation would carry over to the agreement in the topic identification annotation. Therefore, we opted for the segment level setup. No particular order of the segments was imposed, but the annotators were recommended to sequentially annotate the segments following the order they appear in the documents. Also, no target number of topics was provided. A copy of the annotation instructions is in Appendix D.

The annotations were performed in 9 documents with 55 segments from the 'Introduction to Kinematics' subject ($L02$) by two annotators (A2 and A3 from the segmentation annotation task). The modality distribution is as follows: 2 HTML (8 segments), 2 PPT (17 segments), 2 PDF (17 segments), and 3 video (13 segments) documents.

3.4.1 Inter-Annotator Agreement Metrics

Given the previously described clustering setting, we use the standard overlapping clustering metric B^3 to evaluate inter-annotator agreement (Amigó et al., 2009). A similar setup was used to assess agreement of Wikipedia articles topic clustering (Ahn et al., 2011), but studies comparing topic identification of segments provided by human judges, to the best of our knowledge, do not exist. Contrary to other standard metrics

```
==========================================
2: Kinematics: Describing motion.
Our first goal is understanding the motion of objects.
The first step is simple: merely DESCRIBING the motion of things.
1) We’ll only talk about “particles”: point like objects, whose structure is
irrelevant.
2) We’ll work in one dimension, e.g. a train moving back and forth on a
straight track.
To describe motion, we need a few basic concepts, quantities, and definitions.
We’ll use English language words but define them mathematically when
possible.
You’ll see that words like ”velocity, acceleration, force, energy, momentum
(which are often sloppy), are, in physics, totally distinct and well defined.
========================================== (A1, A2, A3)
1) POSITION: Where is the object?
You need a reference frame to describe position.
A reference frame means a choice of axis and coordinate system: where is the
origin, what units will we use to measure length, which direction will we call
positive?
It’s a convention, YOU choose the reference frame.
In 1-D horizontal motion, I will usually pick an origin, and let the positive
direction be to the right, like in a number line.
Position has a SIGN in 1-D: x=+2.5 and x=-2.5 are totally different positions.
2) TIME: When does an event occur?
You need a reference frame here too: when do you define ”t=0” to occur?
I label time by ”t”, which is an INSTANT or POINT in time.
3a) DISPLACEMENT: This is the net CHANGE in position.
x = +2 m : the object has moved 2 meters to the right
The Greek letter there is a ”Delta”, it always means ”change” in this class.
x = -2 m means something different, the object has moved 2 meters to the left.
3b) DISTANCE.
The total length of the path the object has traveled.
It’s different from displacement in several ways.
It’s a positive number, a scalar.
If an object moves forward 2 meters and then back 2 meters, the DISTANCE
traveled is 4 meters, but the displacement is zero!
Mathematically: x = x final - x initial
Position and displacement are useful, but when describing motion, you often
care about more, e.g. how fast it’s moving.
========================================== (A1, A3)
4a) AVERAGE SPEED = (distance traveled)/(time taken).
This is always +, it’s called a scalar.
In the previous example, if we started at t=0, and then point x2 was reached at
20s, point x3 at 30s, and the end was at 60s, then average speed = (60 m)/(60
sec) = 1 m/s.
========================================== (A1, A2)
```

Figure 3.3: Excerpt of an annotated PDF document. At the end of each segment boundary line, we indicate who were the annotators that marked the boundary.

(Purity, Inverse Purity, Entropy, Folkes and Mallows, and Rand Index), the B^3 metric has been shown to comply with four crucial constraints that characterize clustering similarity (Amigó et al., 2009):

- **Cluster homogeneity**: clusters must be homogeneous, i.e., they should not mix items belonging to different classes.
- **Cluster completeness**: items belonging to the same class should be grouped in the same cluster.
- **Rag Bag constraint**: adding items to a bad cluster is less harmful than adding items to a good cluster.
- **Cluster Size vs. Quantity**: a small error in a big cluster should be preferable to a large number of small errors in small clusters.

The B^3 metric decomposes the clustering evaluation in item-wise Precision (Pre) and Recall (Rec) (Figure 3.4). Precision represents the fraction of items within a cluster that belong to the cluster's class. The recall of an item represents how many items within the item's class appear in the cluster. The overall B^3 precision and recall are the averaged precision and recall of all items in the clustering. The final B^3 metric is obtained by combining precision and recall:

$$B^3 = \frac{1}{\alpha(\frac{1}{Pre}) + (1-\alpha)(\frac{1}{Rec})} \tag{3.6}$$

In our experiments, we set $\alpha = 0.5$, which corresponds to the F_1 score version of B^3.

We also used the k_{Fleiss} score, described in Section 3.3.1, since it is a more standard agreement metric. In order to use k_{Fleiss}, it is necessary to provide a list of individual items for the annotators to judge rather than the raw clustering. This was done by computing all pairwise combinations of segments.

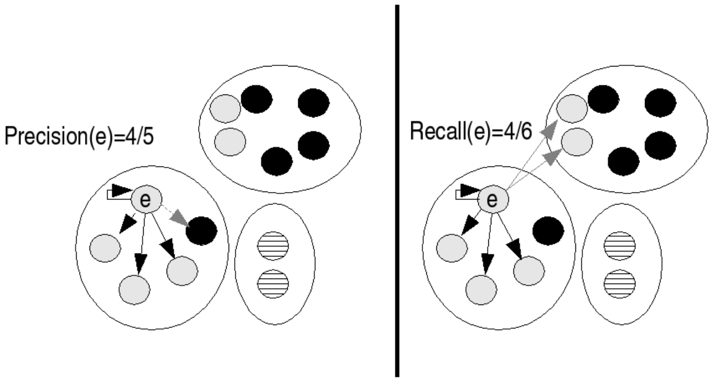


Figure 3.4: Example of calculating the B^3 precision and recall of a point (Amigó et al., 2009).

3.4.2 Inter-Annotator Agreement Results

The inter-annotator agreement results (Table 3.10) show that the annotators agreed reasonably on the topic identification of segments. For the B^3 metric there are no guidelines regarding what high inter-annotator agreement values are, and, thus, we compare it with a baseline where segments are randomly assigned to topics. We obtained $B^3 = 0.84$, a 0.53 increase over the baseline, indicating that the overall clustering of the two annotators was similar. Regarding the k_{Fleiss} score, a high agreement value was obtained (0.78). Despite indicating that annotators had a high agreement, this result must be interpreted with caution. Looking at the confusion matrix (Table 3.11) we can see that there is a class imbalance problem since most of the items correspond to pairs of segments for which the annotators agreed they should not belong to the same topic. We can see the effects of this imbalance when looking at the raw agreement percentage, which drops from 96.4% to 66.7% when discarding the true positive cases. Discarding the true positives is not an ideal solution since the disagreement cases would bias the coefficient towards a high disagreement. Because the clustering setting was translated into an item classification setting, the number of disagreement cases is high since a single segment topic difference will generate multiple disagreement items (one for each other segment in that topic).

k_{Fleiss}	B^3	$B^3_{baseline}$
0.78	0.84	0.31

Table 3.10: Inter-annotator agreement for topic identification.

A3 A2	+	-
+	106	26
-	27	1326

Table 3.11: Topic identification confusion matrix.

We also investigate what is the impact of modality in the topic identification annotation task. That is, if we observe differences when comparing the agreement in segments with the same (intra-)modality or with different (inter-)modality (Table 3.12). In general, the PPT modality obtained the lowest results in both metrics and in both inter and intra-modality segments. The difference ranges between 0.12 and 0.37, for k_{Fleiss}, and 0.04 and 0.13, for B^3. This indicates that PPT documents are harder to annotate. The results in other segment modality combinations were much closer. In these cases, the highest B^3 differences are between HTML and HTML-PPT, and Video and PPT-Video (0.09 difference).

	k_{Fleiss}	B^3
HTML	0.88	0.94
PPT	0.56	0.81
PDF	0.88	0.94
Video	0.78	0.88
HTML-PPT	0.93	0.85
HTML-PDF	0.76	0.92
HTML-Video	0.84	0.90
PPT-PDF	0.77	0.85
PPT-Video	0.68	0.79
PDF-Video	0.81	0.88

Table 3.12: Inter-annotator agreement for modality-based topic identification.

3.4.3 Qualitative Analysis

We now carry out a qualitative analysis of the segment topic identification in order to study possible agreement/disagreement patterns on an empirical basis. In Tables 3.13 and 3.14, we provide the topic descriptions and how many segments where assigned to that topic by each annotator. The number of identified topics is similar between the annotators (17 and 18, for A2 and A3, respectively). From the topic descriptions, we can observe that, in general, the annotators agreed on which topics are described in the segments. One noticeable difference is the exercise topics which only exist in the annotations from A3. For the ‘Example Exercises‘ topic there is a correspondence in A2’s annotations with the ‘Position Time Plot’ topic; another example of different topic descriptions for the same segment cluster is the ‘Average Velocity’ (A2) and ‘Velocity’ (A3) topics. The ‘Examples Exercises Free Fall Gravity‘ case is different since it is not just a matter of assigning different topic descriptors. This topic together with ‘Free Fall Gravity’ corresponds to A2’s single cluster ‘Free Falling Objects’. Therefore, we can see that A3 made a deliberate decision in separating exercise-related segments from more descriptive ones, while A2 put them together on the same topic. The reason for these differences is that the exercise segments include detailed descriptions of the concepts needed to solve the exercises.

Analyzing the segments annotated with more than one topic also allows understanding the topic identification process of the annotators. A2 annotated 11 segments with more than one topic, where 9 segments had 2 topics, and the remaining 2 had 3 topics. This topic overlapping is related to the coupling of related concepts in the same segments. One example is the average and instantaneous velocity concepts for which some segments describe them in isolation while in others they are described in a coupled manner, which is a way to emphasize to students that these are distinct concepts. This makes us conclude that some degree of decoupling between segmentation and the topics present is possible. Therefore, the same vocabulary used

Table 3.13: A2 topic identification annotations.

Topic Description	#Segs
Reference Point	2
Motion Rest	1
Displacement	5
Distance Traveled	3
Velocity and Speed Definition	1
Instantaneous Velocity	5
Instantaneous and Average Speed	6
Average Velocity	6
Position Time Plot	5
Acceleration Definition	6
Instantaneous Acceleration	5
Average Acceleration	7
Acceleration Direction	1
Free Falling Objects	6
General Equation of Motion	5
Velocity Plot Constant Acceleration	3
Circular Motion	1

Table 3.14: A3 topic identification annotations.

Topic Description	#Segs
Motion	1
Reference Point	1
Displacement	6
Velocity	6
Instantaneous Velocity	3
Speed	2
Speed Average	6
Acceleration Definition	8
Instantaneous Acceleration	5
Average Acceleration	4
Constant Acceleration	2
Formalization Equations	7
Free Fall Gravity	4
Graphical Representation	2
Circular Motion	1
Example Exercises	6
Exercises Free Fall Gravity	3
Exercise Average Velocity	1

to describe two different topics can be interwoven in such a way that it can originate a single segment or two distinct segments. The topic overlapping annotations from A3 presents the previous patterns as well. A3 has 9 different segments with more than 1 topics (7 segments with 2 topics, 1 with 3 topics, and 1 with 4 topics). The main difference is again in the exercise-related topics where A3 related those segments with the descriptive counterparts, which did not happen with A2 since they were all in one cluster.

Despite the previously identified differences, the overall qualitative analyses is inline with the inter-annotator agreement scores and corroborates the hypothesis that it is indeed possible for human judges to agree on topic identification judgments between different segments. Given the inter-annotator-agreement results and the analyses presented in this chapter, we conclude that the collected dataset is suitable to carry out a performance evaluation of the models we propose in Chapter 4 in the multi-document segmentation and topic identification tasks.

4 Proposed Solution

Before starting to describe the proposed solution, we provide a general background overview on Bayesian probabilistic modeling needed in Section 4.1 to put in context the remaining sections. To test the hypothesis that a joint model is an effective approach to multi-document segmentation and topic identification, we want to compare it with other approaches that perform the tasks in a pipeline fashion. In this context, we extend the Bayesseg and PLDA models to the multi-document segmentation case in a non-joint model approach (Section 4.2) and also propose a graph-community approach to the topic identification (Section 4.3). Finally, in Section 4.4, we describe our main proposed approach, BeamSeg, which jointly models multi-document segmentation and topic identification by assuming that vocabulary usage relationships between segments exists and that topics are shared across different documents.

4.1 Bayesian Modeling Background

In the following sections, we provide a high-level overview of the Bayesian perspective on probability.

4.1.1 Hypothesis Estimation

The probabilistic graphical models framework (Koller and Friedman, 2009) allows to encode independence relationships between random variables. By analyzing the structure of the graphical model, one can quantify the probability of a hypothesis z given observed data $\mathcal{X}$, $p(z|\mathcal{X})$. With the specification of the model and observed data, we can perform inference to choose the 'best' hypothesis, according to some definition of best. The definition of z is application dependent and corresponds to the unobserved variables of the model. For example, in LDA, hypotheses are the parameterizations of word probability distributions and word topic assignments. In this thesis, we are interested in evaluating different segmentation hypothesis. Following a Bayesian approach, we resort to Bayes rule to obtain the expression for $p(z|\mathcal{X})$:

$$p(z|\mathcal{X}) = \frac{p(\mathcal{X}|z)p(z)}{p(\mathcal{X})} \tag{4.1}$$

Equation 4.1 is described in literature as the *posterior* distribution of $p(z|\mathcal{X})$. The numerator of the expression is referred to as the *joint* probability distribution, and its form depends on the structure of the probabilistic graphical model. It is composed of two terms, the *likelihood*, $p(\mathcal{X}|z)$, and the *prior* $p(z)$. The likelihood determines how good z is at explaining the observed data $\mathcal{X}$, and the prior expresses how well z matches our expectations of what a good hypothesis looks like. In this context, the prior acts like a learning bias which we can tune to have a stronger or weaker influence on the probability of $p(z|\mathcal{X})$. This determines the balance between the observed data and the prior. A strong prior will require more evidence in the data to contradict our pre-existing belief of what a good hypothesis is. A weak or non-informative prior, in the sense that it does not favor any particular type of hypothesis, requires few evidence to accept a hypothesis.

One procedure to learn to select the best hypothesis z is maximum likelihood estimation (MLE). As the name implies, the procedure chooses the hypothesis that maximizes the likelihood term:

$$\tilde{z}_{\mathrm{MLE}} = \operatorname*{argmax}_{z} p(\mathcal{X}|z) \tag{4.2}$$

To illustrate how MLE works, we will use the standard coin flipping example. Consider we observe a coin flip sequence $\mathcal{X} = \mathrm{HHHTT}$ (H = heads and T = tails) and we want to find a good estimate for the probability of getting heads, z. In a MLE scenario, this means we are looking for the hypothesis value $\tilde{z}_{\mathrm{MLE}}$ that is most likely to have generated the observed data. This simply amounts to performing counts on the data and normalize the probabilities to sum to one, yielding $\tilde{z}_{\mathrm{MLE}} = \frac{3}{5}$, in this example. It should be noted that the absence of a prior term in MLE is equivalent to assuming that all hypotheses are equally probable.

Another possibility to learn to select the best hypothesis is with the maximum a posteriori (MAP) framework:

$$\begin{aligned}\tilde{z}_{\mathrm{MAP}} &= \operatorname*{argmax}_{z} p(z|\mathcal{X}) \\ &= \operatorname*{argmax}_{z} \frac{p(\mathcal{X}|z)p(z)}{p(\mathcal{X})} \\ &= \operatorname*{argmax}_{z} p(\mathcal{X}|z)p(z), \end{aligned} \tag{4.3}$$

where the denominator can be ignored, since it is a constant across the argmax_z operator. The advantage of MAP is that prior knowledge about the hypothesis can be encoded through $p(z)$. Resuming the previous example, we might have reasons to believe that the coin is rigged and produces more tails than head flips. To express this prior belief, $p(z)$ can be set to favor values $z < 0.5$. The more heavily biased the prior distribution is, the more evidence in the data is necessary to contradict our prior belief that the coin is unfair.

The previous approaches only consider individual point estimates of z in order to decide the most likely hypothesis. This is what the argmax_z operator of Equations 4.2 and 4.3 implies since it can be interpreted as trying all possible z values individually and return the one with the highest likelihood. In contrast, the full posterior is, in fact, a true probability distribution over the possible values of z. This allows computing the expected value of the distribution, which fulfills the goal of incorporating as much information as we have from the posterior in the estimate. The expected value is defined as follows:

$$\mathbb{E}[f(z)] = \sum_{z \in \mathcal{Z}} f(z)p(z) \tag{4.4}$$

$$\mathbb{E}[f(z)] = \int f(z)p(z)dz, \tag{4.5}$$

for the discrete and continuous cases, respectively. $\mathcal{Z}$ is the set of discrete values z can take, and $p(z)$ is the probability distribution over possible values of z. In the coin example, $f(z) = p(y|z)$, and $p(z) = p(z|\mathcal{X})$; where $p(y|z)$ is the probability of prediction y (heads in the coin example) given the hypothesis z. Under these condition we can no longer ignore the denominator $p(\mathcal{X})$ in Equation 4.1, the *evidence* term:

$$p(\mathcal{X}) = \int p(\mathcal{X}|z)p(z)dz \tag{4.6}$$

The evidence term corresponds to integrating over all possible values (or summing in the discrete case). Essentially it acts like a normalizing term for the posterior distribution. When using MAP estimates, we get a kind of unnormalized score for the hypothesis, but what we really want is the normalized version in order to have a true probability distribution to take expected values. Only this way we fully take into account prior knowledge regarding z and the interactions it has with the observed data $\mathcal{X}$. Concluding this line of thought, imagine we get a coin flip sequence $\mathcal{X} = \text{TTT}$. The MLE estimate would be $\tilde{z}_{\text{MLE}} = 0$, since we did not observe any heads. If we use the expected value of a posterior distribution of z, assuming a non-informative prior, we obtain the following expression (for simplicity we omit the derivation details):

$$\mathbb{E}[p(y|z)] = \frac{\#\text{H} + 1}{\#\text{H} + \#\text{T} + 2} \tag{4.7}$$

where $\#\text{H}$ and $\#\text{T}$ are the number of observed heads and tails respectively. Plugging in the observed data, we get $\mathbb{E}[p(y|z)] = 0.2$. Intuitively, this is a better estimate than $\tilde{z}_{\text{MLE}} = 0$, since we would think with such little data we were just unlucky when flipping the coin. The problem of MLE is that it tends to accept extreme data too easily. Despite this toy example, it does illustrate why generally we should prefer to use expected values for hypothesis estimation.

4.1.2 Inference with Gibbs Sampling

As we concluded in the previous section, a better approach to inference should use the full posterior distribution. In practice, there is one caveat to being able to do this, which is the computation of the evidence term $p(\mathcal{X})$. Models used in real applications are too complex to afford an analytical solution to $p(\mathcal{X})$, such as the one for the coin flip toy problem. This motivates the approach to inference using Gibbs sampling (Griffiths and Steyvers, 2004; Robert and Casella, 2005; Papanikolaou et al., 2017; Terenin et al., 2019), which gives assess to the posterior while bypassing the problem of computing the evidence term. The intuition for Gibbs sampling is that we sample from a distribution that asymptotically follows $p(z|\mathcal{X})$, which does not require $p(\mathcal{X})$ to be computed. This is done by sampling points from $f(z)$ according to $p(z|\mathcal{X})$. After gathering N samples, we can compute the expectation as follows:

$$\mathbb{E}[f(z)] = \lim_{N\to\infty} \frac{1}{N}\sum_{t}^{N} f(z^t), \tag{4.8}$$

which resembles the average value of the collected samples. This is similar to the definition of the expected value, without $p(z)$. In fact, $p(z)$ is implicit since we assume the samples are collected according to $p(z|\mathcal{X})$. This allows collecting samples from the regions of posterior with the highest probability. Therefore, for the Gibbs sampling approach to be effective, it is necessary to ensure we spend more time collecting samples from such regions of interest.

Now we need to define how to sample according to $p(z)$. Different approaches to address this problem are possible, such as rejection sampling, adaptive rejection sampling, Metropolis-Hasting, and importance sampling (Bishop, 2006). We will focus on the Gibbs sampling approach, an instance of a Markov Chain Monte Carlo (MCMC) algorithm (Robert and Casella, 2005). In this framework, the z's are viewed as points in a state space that we want to explore in such way that the likelihood of visiting some z is proportional to $p(z)$. In this context, we are making probabilistic choices about which z to visit next, the Monte Carlo part of the algorithm. The Markov Chain part stems from assuming that next visited state, z^{t+1}, only depends on the current state, z^t. In Gibbs sampling instead of choosing a state out of all possible states, we change one variable from z^t, while the remaining ones are fixed. For example, a Gibbs sampler for LDA would not choose a state out of all possible combinations of topic assignments, instead, it would choose one-word topic assignment from z^t. Samples from the target distribution are obtained as follows:

$$z_i^{t+1} \sim p(z_i^t | z_1^{t+1}, ..., z_{i-1}^{t+1}, z_{i+1}^t, ..., z_k^t), \tag{4.9}$$

where z_k corresponds to one of the variables of the model. It should be noted that the indexes in the conditional probability expression define that we are sampling a new value for z_i^{t+1} based on the values of all other variables. Moreover, as soon as a new value is sampled, it is used when sampling the next variable. A full iteration of the Gibbs sampler corresponds to sample all z_k variables, obtaining the new full state sample $f(z^{t+1})$. The process is repeated N times, and the results averaged in the end. As iterations take place, we start getting closer to the region of the posterior distribution we are interested in, and consequently, the samples will get better. Therefore, if N is large enough, we will obtain a good estimate for the parameters of our model. Another important property of the sampling expression (Equation 4.9) is that after applying the conditional probability definition, we obtain in the numerator the expression of the joint probability of the model, and, in the denominator, the joint minus z_i^t. This is how we avoid computing the evidence term since it gets canceled in the expression.

4.1.3 Inference with Variational Inference

In this section, we give a high-level overview of the Variational Inference (VI) approach to the posterior inference problem (Jordan et al., 1999; Wainwright and Jordan, 2008). VI frames inference as an optimization problem, as opposed to a sampling algorithm like Gibbs sampling. The argument that favors the use of VI is scalability since it can be faster than Gibbs sampling. The downside of VI is that it does not enjoy the theoretical properties of Gibbs sampling, in the sense that it is an approximation of the posterior distribution. Under what conditions the VI approximation is good is still an open research question. The main idea in VI is to define a family of variational distributions $\mathcal{Q}$ over the model's latent variables z. Each $q(z) \in \mathcal{Q}$ is parametrized by variational parameters and is a candidate approximation to the exact posterior. Thus, the goal of the optimization is to find the $q^*(z)$ that is closer to the target posterior distribution, the one with the best parametrization. Closeness is measured by the KL divergence to the posterior distribution:

$$q^*(z) = \operatorname*{argmin}_{q(z) \in \mathcal{Q}} \mathrm{KL}(q(z)||p(z|\mathcal{X})) \tag{4.10}$$

Solving the optimization problem in Equation 4.10 is intractable since it still requires to compute the evidence term. This is observed when applying the definition of KL to Equation 4.10:

$$\begin{aligned}
\mathrm{KL}(q(z)||p(z|\mathcal{X})) &= \mathbb{E}\Big[\log \frac{q(z)}{p(z|\mathcal{X})}\Big] \\
&= \mathbb{E}[\log q(z)] - \mathbb{E}[\log p(z|\mathcal{X})] \\
&= \mathbb{E}[\log q(z)] - \mathbb{E}[\log p(z,\mathcal{X})] + \log p(\mathcal{X})
\end{aligned} \tag{4.11}$$

VI addresses this problem by minimizing a related objective function, the evidence lower bound (ELBO). The ELBO is equivalent to the negative KL divergence, differing only by a constant:

$$\mathrm{ELBO}(q) = \mathbb{E}[\log p(z, \mathcal{X})] - \mathbb{E}[\log q(z)] \tag{4.12}$$

To fully specify the optimization problem we need to define the variational distribution family $\mathcal{Q}$. Various possibilities exist (Saul and Jordan, 1995; Salimans and Knowles, 2013), but we will focus on the mean-field variational family since it is the approach we use later. The main assumption in mean-field variational inference is that the latent variables are independent. Thus, they are generally defined as follows:

$$q(z) = \prod_{j=1}^{m} q_j(z_j) \tag{4.13}$$

From Equation 4.13 we see that each z_j has its own individual variational parameters, $q_j(z_j)$, making them independent of each other. Another thing to notice from Equation 4.13 is that it does not model the observed data. The connection to the data only appears when maximizing the ELBO, through the $\mathbb{E}[\log p(z, \mathcal{X})]$ term. Equation 4.13 defines a generic mean-field variational distribution. Applying this to an actual model requires the specification of a parametric form. For example, in LDA, we can define Categorical variational parameters for the topic assignments since these are Categorical as well.

With the previous setup in place, the missing piece is how to actually perform the optimization task. Again, different approaches to this problem exist (Hoffman et al., 2010; Ranganath et al., 2014; Srivastava and Sutton, 2017; Zhu et al., 2018; Chien and Lee, 2018), but we will only give some intuition for the coordinate ascent mean-field variational inference (CAVI) approach. CAVI is an iterative algorithm which optimizes each variational parameter individually while holding the others fixed. This is done by taking the gradient of the ELBO with respect to an individual variational parameter $q_j(z_j)$, set it equal to zero, and solve for the new value. An iteration of the algorithm corresponds to making a pass at all $q_j(z_j)$ parameters and update them. At the end of each iteration, the ELBO is computed to monitor convergence. This procedure corresponds to going uphill on the ELBO until a local minimum is found.

4.2 Extending Single-Document Segmentation Models

In this section, we extend Bayesseg and PLDA (both described in Section 2.2.2) to multi-document segmentation. The goal is to study how approaches only modeling multi-document segmentation perform.

4.2.1 Bayesseg for Multi-Document Segmentation

We start by describing the original Bayesseg model, a probabilistic approach to single-document segmentation, where the core idea is to assume that segments with the same topic are drawn from the same Categorical language model[1]. The formalization of the generative process is as follows:

1. For each utterance $u \in \{1, ..., U\}$ in d:
   (a) If $s_u \neq s_{u-1}$ draw word distribution:
   $\phi_{s_u} \sim \text{Dirichlet}(\beta)$.
   (b) Draw words:
   $\mathbf{x}_u \sim \text{Categorical}(\phi_{s_u})$

The previous process defines that all u utterances in a segment s have their bag-of-words representation $\mathbf{x}_u$ drawn from a Categorical language model ϕ_{s_u}; where s_u is the hidden segment assignment variable of u. Language models are drawn from a Dirichlet prior parametrized by β. The model constrains segmentations to yield linear segmentations. This induces higher likelihood segmentations to have language models concentrating probability mass on a small subset of the vocabulary. Conversely, low likelihood segmentations spread the probability mass on a broader set of words. This modeling behavior is attuned with the lexical cohesion theory.

Using a Dirichlet prior in the previous setup allows us to encode assumptions about how language models should look like. We expect the segment language models to be sparse, meaning that only a small subset of the words has a high probability. This goes hand in hand with the lexical cohesion assumption where topic segments tend to heavily favor some part of the vocabulary. By appropriately setting the β parameters, it is possible to achieve this behavior. Another reason for using a Dirichlet prior is the fact that it is conjugate to the Categorical distribution (the result of multiplying both distributions is also Dirichlet distributed). As we will see in the derivations below, this has mathematical convinces that cannot be achieved with other similar distributions such as the Logit-normal distribution (Atchison and Shen, 1980).

After having fully specified the model, we define the joint likelihood is as follows[2]:

$$p(\mathbf{X}|S, \Phi, \beta) = \prod_{u=1}^{U} p(\mathbf{x}_u|\phi_{s_u})p(\Phi|\beta), \tag{4.14}$$

[1] A summary of the mathematical notation can be found in Appendix A, Table A.1

[2] We adopt the notational convention where bold variables correspond to vectors.

where $\mathbf{X}$ is the set of bag-of-words representations of all U utterances in a document, Φ is the set of hidden language models, and S is a set of variables that relates $\mathbf{x}_u$ to its corresponding segment, ϕ_{s_u}. To avoid searching the full space of language models, a marginalizing process is carried out by appealing to conjugacy between the Categorical language models and the Dirichlet prior. This allows the conjugate Dirichlet distribution to integrate to one, leaving the marginalized joint likelihood expression with the normalizing constants. Using the probability density function definition of a Dirichlet distribution,

$$\text{Dirichlet}(\beta) = \frac{\Gamma(\sum_{i=1}^{W} \beta_i)}{\prod_{i=1}^{W} \Gamma(\beta_i)}, \tag{4.15}$$

the derivation for marginalizing language model ϕ from segment s is as follows:

$$\begin{aligned}
p(\mathbf{X}|s,\beta) &= \int p(\mathbf{X}|s,\phi)p(\phi|\beta)d\phi \\
&= \int \prod_{w=1}^{W} \phi_w^{n_w^s} \Big(\frac{\Gamma(W\beta)}{\Gamma(\beta)^W}\Big) \prod_{w=1}^{W} \phi_w^{\beta-1} d\phi \\
&= \int \Big(\frac{\Gamma(W\beta)}{\Gamma(\beta)^W}\Big) \prod_{w=1}^{W} \phi_w^{n_w^s+\beta-1} d\phi \\
&= \frac{\Gamma(W\beta)}{\Gamma(\beta)^W} \frac{\prod_{w=1}^{W}\Gamma(n_w^s+\beta)}{\Gamma(\sum_{w=1}^{W} n_w^s + W\beta)} \int \frac{\Gamma(\sum_{w=1}^{W} n_w^s + W\beta)}{\prod_{w=1}^{W}\Gamma(n_w^s+\beta)} \prod_{w=1}^{W} \phi_w^{n_w^s+\beta-1} d\phi \\
&= \frac{\Gamma(W\beta)}{\Gamma(\beta)^W} \frac{\prod_{w=1}^{W}\Gamma(n_w^s+\beta)}{\Gamma(\sum_{w=1}^{W} n_w^s + W\beta)},
\end{aligned} \tag{4.16}$$

where W is the size of the vocabulary, n_w^s is the count of word w in s, and Γ corresponds to the Gamma function. Although Equation B.6 only accounts for a single segment, it can be easily extended to provide the joint likelihood of the full document segmentation by applying it to each segment and multiplying the individual results. During inference Bayesseg finds the segmentation S that maximizes the likelihood of the joint distribution of the model. Therefore, inference amounts to finding the segmentation $\hat{S} = \text{argmax}_S \; p(\mathbf{X}|S,\beta)$, a MAP approach under a uniform prior.

In the previous setup, the language models are marginalized out and since only one document is given as input an exhaustive exploration of the segmentation state space is possible. This is done using a matrix (Figure 4.1) where each (u_i, u_j) entry corresponds to the likelihood of a segment that begins on the u_i utterance and ends at u_j. For example, entry (u_2, u_4) contains the likelihood value obtained by using Equation B.6 considering just u_2, u_3, and u_4. The values in the matrix correspond to the joint likelihood of a single segment hypothesis. To find the best overall segmentation, a dynamic programming approach is used. The algorithm works by finding the best segmentations up to an utterance u_i. In each iteration, the next

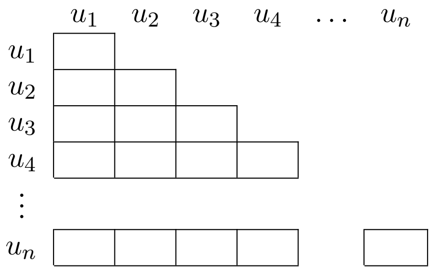


Fig. 4.1: Segment likelihood matrix.

utterance is added and the best segmentation is found for the augmented set of utterances. The algorithm keeps track of the best segmentations at each iteration since they are needed in subsequent iterations. This is the dynamic programing aspect of the procedure. The first iteration of the algorithm computes the value for the (u_1, u_1) entry, which corresponds to the likelihood of having a segment with just utterance u_1. Obtaining this value is trivial since only one segmentation is possible. The next iteration computes the best segmentation up to utterance u_2 using the second line of the matrix. Since there are two entries on this line, two segmentations are possible: one with a segment containing u_1 and u_2 (segment s_{1-2}), and another with a segment containing u_2 (segment s_{2-2}) plus the best previous segmentation. The likelihood of the first segmentation corresponds to the value on the (u_2, u_1) entry. The likelihood value for the latter case corresponds to the sum of the (u_1, u_1) and (u_2, u_1) entries. The algorithm terminates when the last entry of the matrix is reached, meaning that the best segmentation for the full document was found. The final segmentation is decoded by backtracking the highest likelihood points.

Having described the original Bayesseg, we now detail our extended approach to include information from other documents based on a lexical similarity approach, the Bayesseg-MD algorithm. One of the difficulties in Bayesseg is that it computes likelihoods for language models using few data since segments generally do not contain many sentences. To address this problem, we add the counts from similar utterances from other documents in the collection. The underlying assumption is that similar sentences are likely to come from the same language model, and, thus, using them can help obtain better segment likelihood estimates. The most similar utterances are chosen according to the following recursive function:

$$U_{sm}(s_{i,j}) = \begin{cases} \underset{u' \in D}{Nmax} \cos(u_{s_i}, u') & \text{, if } i = j \\ \underset{u' \in D}{Nmax} \cos(u_{s_i}, u') \cup U_{sm}(s_{i+1,j}) & \text{, if } i < j \end{cases} \tag{4.17}$$

where $s_{i,j}$ is a segment spanning from the i^{th} to the j^{th} utterances in a target document (with $i \leq j, \forall_{i,j}$), u_{s_i} is the i^{th} utterance from segment $s_{i,j}$, u' is an utterance from the set of documents in dataset D, and

$Nmax$ is an operator that returns the top N arguments that maximize some function, in our case the cosine similarity function between two utterances. Using this function we rewrite Equation B.6 as follows:

$$p(\mathbf{X}|s_{ij},\beta) = \frac{\Gamma(W\beta)}{\Gamma(\beta)^W} \frac{\prod_{w=1}^{W} \Gamma(n_w^s + n_w^{U_{sm}(s_{i,j})} + \beta)}{\Gamma(\sum\limits_{w=1}^{W} n_w^s + n_w^{U_{sm}(s_{i,j})} + W\beta)}, \tag{4.18}$$

where i and j are the first and last sentences in segment $s_{i,j}$, and $n_w^{U_{sm}(s_{i,j})}$ are the counts of word w in the set of utterances given by $U_{sm}(s_{i,j})$. It should be noted that not all words from $U_{sm}(s_{i,j})$ are added to the segment word counts, only the top-40% highest *tf-idf* values. The goal is to use only words that are relevant for all documents in D. This avoids introducing noise from words that are too document or modality specific.

4.2.2 PLDA for Multi-Document Segmentation

Similarly to before, we now describe how we have extended PLDA (Purver et al., 2006), to a multi-document segmentation model, which we refer as PLDA-MD[3].

In the original PLDA, each utterance u in the document is associated with a binary switching variable c_u. Segmentation is defined by the sequence of all $\mathbf{c}$ variables. For example, if $\mathbf{c} = (0,0,0,1,0,1,0)$, three different segments exist, with the utterances aggregated in the following way: $\{1,2,3\},\{4,5\},\{6,7\}$. The probability of starting a new segment is defined as π, $p(c_u = 1) = \pi$. Associated to utterance u is a topic proportions variable θ_u. Utterances belonging to the same segment have the same topic proportions. Therefore, if $c_u = 0$, then $\theta_u = \theta_{u-1}$. When a new segment starts, $c_u = 1$, new topic proportions θ_u are drawn from a Dirichlet prior.

In our PLDA-MD extension, we need to consider that the variables need to take into account a collection of documents, D, rather than a single document. In this context, we now have $c_{d,u}$ topic shift variables for each u utterance in each document $d \in D$. The probability of starting a new topic is now also per document, π_d. This makes a modeling assumption that different documents can have different expected segment lengths, which is a reasonable assumption given that documents can have different modalities. For example, textbooks are much more verbose than presentation slides, and, thus, their segments should be lengthier as well. Each segment in each document has its own $\theta_{d,u}$ segment topics proportions variable. The last modification is to assume, now, that for each k in a set of topics K, the corresponding ϕ_k language

[3] A summary of the used mathematical notation can be found in Appendix A, Table A.2.

models are shared among all documents. Given the previous setup, an utterance u is generated by sampling a topic assignment $z_{w_{u,i}}$ for each word $w_{u,i}$ in u, according to topic proportions $\theta_{d,u}$. Depending on the drawn topic, $w_{u,i}$ is sampled from the corresponding topic $\phi_{z_{w_{u,i}}}$. The π_d and ϕ_k variables are generated from Beta and Dirichlet priors, with γ and β parameters, respectively. Having defined all variables of the model, we obtain the corresponding joint distribution and dependency structure depicted in Figure 4.1. The summary of the generative model of PLDA-MD is:

1. For each topic $k \in \{1, ..., K\}$, draw word distribution $\phi_k \sim \text{Dirichlet}(\beta)$.

2. For each document $d \in \{1, ..., D\}$,

    (a) Draw topic segment probability $\pi_d \sim \text{Beta}(\gamma)$.

    (b) For each utterance $u \in \{1, ..., U\}$ in d:

      i. Draw segment indicator:
      $c_{d,u} \sim \text{Bernoulli}(\pi_d)$

      ii. Draw topic proportions:
      $\theta_{d,u} \sim \text{Dirichlet}(\alpha)$, if $c_{d,u} = 1$, otherwise $\theta_{d,u} = \theta_{d,u-1}$.

    (c) For each word $w_{u,i} \in \{1, ..., W_u\}$ in u,

      i. Draw topic:
      $z_{w_{u,i}} \sim \text{Categorical}(\theta_{d,u})$

      ii. Draw word:
      $w_{u,i} \sim \text{Categorical}(\phi_{z_{w_{u,i}}})$

In Figure 4.1, it is possible to observe that the D plate encodes that the same language models generate the segments from all documents in the dataset, meaning that they are shared across segments in different documents. It also makes explicit that each document has an individual expected segment length. These are the two multi-document aspects of PLDA-MD.

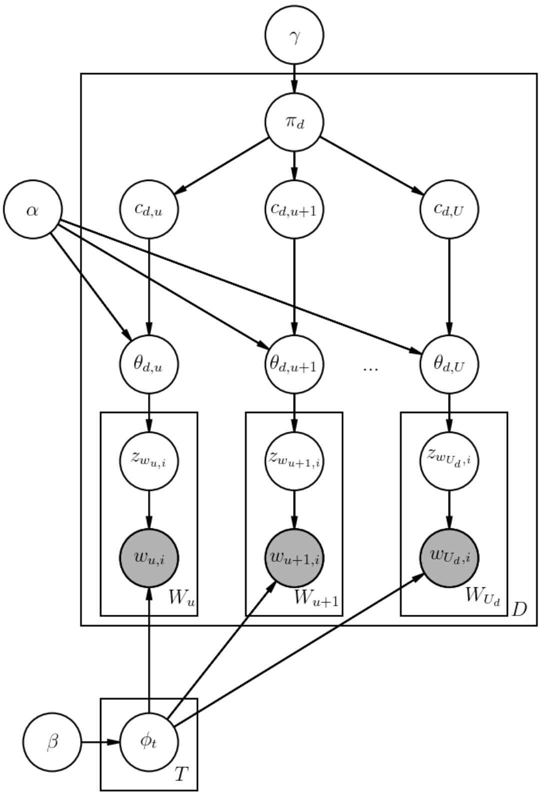


Figure 4.1: Plate diagram for the PLDA-MD model.

4.2.2.1 Inference

Given the PLDA-MD model specification, inference is carried out by evaluating the posterior distribution to find the parameter configuration that best explains the observed data $\mathbf{w}$. Our approach to inference resorts to Gibbs sampling. The complete derivations for the Gibbs sampler can be found in Appendix B, as we only present a summary of the approach in this section.

We start by simplifying the posterior distribution by integrating out some of the parameters, namely π, θ, and ϕ. The remaining variables are the topic assignments $\mathbf{z}$ and the topic shift variables $\mathbf{c}$. By applying Bayes rule we obtain the following expression for the posterior:

$$p(\mathbf{z}, \mathbf{c}|\mathbf{w}) = \frac{p(\mathbf{w}|\mathbf{z})p(\mathbf{z}|\mathbf{c})p(\mathbf{c})}{\sum_{\mathbf{z},\mathbf{c}} p(\mathbf{w}|\mathbf{z})p(\mathbf{z}|\mathbf{c})p(\mathbf{c})} \tag{4.19}$$

Integrating out π gives $p(\mathbf{c})$ the following expression:

$$\begin{aligned} p(\mathbf{c}) &= \int p(\mathbf{c}|\pi)p(\pi|\gamma)d\pi \\ &= \left(\frac{\Gamma(2\gamma)}{\Gamma(\gamma)^2}\right)^D \prod_{d=1}^{D} \frac{\Gamma(n_1^d + \gamma)\Gamma(n_0^d + \gamma)}{\Gamma(N_d + 2\gamma)}, \end{aligned} \tag{4.20}$$

where n_x^d is number of $c_{d,u} = x$ variables in d, and N_d is the number of segments in d. Similarly, we derive the expression for $p(\mathbf{w}|\mathbf{z})$:

$$\begin{aligned} p(\mathbf{w}|\mathbf{z}) &= \int p(\mathbf{w}|\mathbf{z}, \phi) p(\phi|\beta) d\phi \\ &= \left(\frac{\Gamma(W\beta)}{\Gamma(\beta)^W} \right)^K \prod_{k=1}^{K} \frac{\prod_{w=1}^{W} \Gamma(n_{D,w}^k + \beta)}{\Gamma(n_D^k + W\beta)}, \end{aligned} \tag{4.21}$$

where $n_{D,w}^k$ is the number of times word w was assigned topic k in the document collection D, and n_D^k is the frequency of topic k in D. Finally, evaluating $p(\mathbf{z}|\mathbf{c})$ results in:

$$\begin{aligned} p(\mathbf{z}|\mathbf{c}) &= \int p(\mathbf{z}|\theta) p(\theta|\mathbf{c}, \alpha) d\theta \\ &= \left(\frac{\Gamma(K\alpha)}{\Gamma(\alpha)^K} \right)^{n_1^D} \prod_{d=1}^{D} \prod_{u \in U_{d,1}} \frac{\prod_{k=1}^{K} \Gamma(n_{d,S_u}^k + \alpha)}{\Gamma(n_{d,S_u}^{\cdot} + K\alpha)}, \end{aligned} \tag{4.22}$$

where n_1^D is the total number of segments in D, $u \in U_{d,1}$ is the set of utterances such that $c_{d,u} = 1$, n_{d,S_u}^k is the frequency of topic k in the segment S_u of document d, and $n_{d,S_u}^{\cdot}$ is the total number of words in S_u.

By combining Equations 4.20, 4.21, and 4.22, we obtain the joint distribution expression of the model, which corresponds to the numerator of Equation 4.19. The problem is the intractable sum in the denominator of the equation. To address this problem we resort to Gibbs sampling. To build the Gibbs sampler we need to derive sampling equations for the $\mathbf{z}$ and $\mathbf{c}$ latent variables. This is done by sampling a single variable given all the remaining ones. We start by defining how to sample a topic assignment variable $z_{d,w_{u,i}}$ given all other $\mathbf{z}_{\neg(d,w_{u,i})}$, and $\mathbf{c}$ variables:

$$p(z_{d,w_{u,i}} = k | \mathbf{z}_{\neg(d,w_{u,i})}, \mathbf{c}, \mathbf{w}) = \frac{n_{D,w_{u,i}}^k + \beta}{n_D^k + W\beta} \frac{n_{d,S_u}^k + \alpha}{n_{d,S_u}^{\cdot} + K\alpha}, \tag{4.23}$$

where all counts in the n terms exclude $z_{d,w_{u,i}}$. To derive Equation 4.23 we take advantage of the fact that $p(\mathbf{c})$ remains the same when excluding $z_{d,w_{u,i}}$, thus, it cancels out, leaving only two factors from Equations 4.21 and 4.22. The final Equation 4.23 has two factors that determine the sampling probability of a topic assignment. The first factor expresses that a word $w_{u,i}$ is more likely to be assigned topic t if we observe similar assignments in D. The second factor pushes the topic assignments towards topics that are more frequently seen in the segment where $w_{u,i}$ belongs to.

The $c_{d,u}$ variables determine if a new segment is going to start. When sampling $c_{d,u}$ given $\mathbf{c}_{\neg(d,u)}$, $\mathbf{z}$, and $\mathbf{w}$, we need to consider the merging and splitting of segments separately. The expression we obtain is in Equation 4.24, where S_u^x is the resulting segmentation when considering $c_{d,u} = x$, and all counts exclude $c_{d,u}$. To derive this expression it is crucial to note that $c_{d,u}$ only affects the S_u segment, and, thus, everything else cancels out. In the split case, a new segment is introduced, hence the S_{u-1}^1 factor in the expression. From Equation 4.24 we can see that sampling $c_{d,u}$ in the context of a document collection D only depends on segment boundaries counts and word topic assignments in d. Therefore, the multi-document aspect of PLDA-MD only affects the sampling of $\mathbf{z}$. Also, when sampling $c_{d,u}$ two factors come into play. The first one indicates how likely $c_{d,u} = x$ is, considering only the distribution of the other $\mathbf{c}$ labels in d. This means that the more non-boundary utterances we have, the more this factor tends to push $c_{d,u}$ to 0. The second factor works as a measure of how much the segmentation S_u^x "likes" $c_{d,u}$. A merge is more likely to occur ($c_{d,u} = 0$) if the words of the resulting segment better fit the underlying topic distributions of the segments. This is analogous for the split case.

$$p(c_{d,u} = x | \mathbf{c}_{\neg(d,u)}, \mathbf{z}, \mathbf{w}) = \begin{cases} \frac{n_0^d + \gamma}{N_d + 2\gamma - 1} \frac{\prod_{k=1}^{K} \Gamma(n_{d,S_u^0}^k + \alpha)}{\Gamma(n_{d,S_u^0}^{\cdot} + K\alpha)}, x = 0 \\ \frac{n_1^d + \gamma}{N_d + 2\gamma - 1} \frac{\Gamma(K\alpha)}{\Gamma(\alpha)^K} \frac{\prod_{k=1}^{K} \Gamma(n_{d,S_{u-1}^1}^k + \alpha)}{\Gamma(n_{d,S_{u-1}^1}^{\cdot} + K\alpha)} \frac{\prod_{t=1}^{K} \Gamma(n_{d,S_u^1}^k + \alpha)}{\Gamma(n_{d,S_u^1}^{\cdot} + K\alpha)}, x = 1 \end{cases} \tag{4.24}$$

Given the derived sampling equation, the Gibbs sampler procedure for PLDA-MD consists of going through all $c_{d,u}$ and $z_{d,w_{u,i}}$ variables and sample new values according to Equations 4.23 and 4.24. A full pass on the variables corresponds to one iteration of Gibbs sampler, where a new state (sample) of the chain is obtained. We run the Gibbs sampler for N iteration and average all the collected samples to obtain the final values for the latent variables $\mathbf{c}$ and $\mathbf{z}$.

4.2.2.2 A Cache Scheme for Gibbs Sampling

Gibbs sampling is known for being computationally expensive. This is related to the number of iterations it requires to converge. Also, the cost of each iteration grows with the number of variables we need to sample. This might be not so relevant in single-document segmentation, but it is for multi-document segmentation. The more documents we have in the collection, the bigger are the $\mathbf{z}$ and $\mathbf{c}$ sets. This problem is further aggravated when the Gibbs sampling equations are complex and expensive to compute as well, which is the case of Equations 4.23 and 4.24. In the literature, it is possible to find different strategies to speed up the Gibbs sampling process. One strategy is to use a fixed order of sampling variables to encourage a faster convergence of the Gibbs sampler, as opposed to the standard random order (Chen and H. Ip, 2014; He et al., 2016). Another strategy is to keep track of all already computed sampling probabilities in a cache structure to avoid complex computations (Bidyuk and Dechter, 2006; Terenin et al., 2015). In this context, we propose a novel caching scheme that combines both ideas. We use a fixed order for sampling the $\mathbf{z}$ and $\mathbf{c}$ variables that stimulates hits in a small-sized cache. This allows us to overcome the limitations of the previous work, where the overhead of querying and updating the cache is considerable.

From the definition of Equation 4.22 it is possible to see that we only require the topic counts of word $w_{u,i}$ in D and in S_u. Therefore, when sampling $w_{u,i}$ under the following conditions:

- $w_{u,i} = w_{u,i-1}$.
- $z_{w_{u,i}} = z_{w_{u,i-1}}$.
- $z_{w_{u,i-1}}$ did not change topic when sampled.
- $w_{u,i}$ and $w_{u,i-1}$ belong to the same segment.

we can use the same topic probabilities of $w_{u,i-1}$. This observation makes the Gibbs sampler lend itself to a caching scheme where the topic probabilities used to sample $z_{w_{u,i}}$ are memorized. To maximize the number of times these conditions are met, we sample $\mathbf{z}$ by word type and order of occurrence in the document collection. By sampling words according to this order, we are increasing the chances of consecutively sampling two words with the same topic, because words from the vocabulary should tend to have the same topic in the same segment. Consequently, we have a higher chance of having a cache hit.

A similar caching approach can also be applied when sampling $\mathbf{c}$. The sampling of $c = 0$ in Equation 4.24 states that we only need to know the topic counts of the corresponding segment and the number of non-boundary utterances in the document. Thus, this sampling probability is the same for two consecutive utterances in the same segment. In this context, we can cache this sampling probability and chose a sampling order of $\mathbf{c}$ according to the utterance sequence in the document. The end result is a sampling scheme that stimulates cache hits since it is likely that a segment is composed of several utterances.

For the previous cache scheme to be effective, it is necessary to impose a fixed order to sample the variables. In the literature, this is defined as a systematic scan order of variables, as opposed to a random scan order. One concern when using this type of strategy is if the Gibbs sampler converges slower when compared with a random scan order. It is still a conjecture whether there is a bound on the convergence time between these two scan orders for a given model (He et al., 2016). Therefore, it is not possible to guarantee that we can interchangeably use them without sacrificing performance. In Section 5.2, we will delve into this issue by empirically analyzing both scan orders under the PLDA-MD model.

4.3 Graph-Community Detection for Topic Identification

The graph-community detection framework allows to find organizational principals in graphs, affording a better understanding of how objects in a graph interact and relate to each other (Backstrom et al., 2006; Yang et al., 2013). This is achieved by grouping nodes in communities. There are various application where determining the community structure of a graph is desirable, for example, in social networks, communities correspond to groups of friends who attended the same school (Leskovec and Mcauley, 2012); in protein interaction networks, communities are functional modules of interacting proteins (Ahn et al., 2010); in co-authorship graphs, communities correspond to scientific disciplines (Girvan and Newman, 2002). In graph-community detection, there are two types of features can be used. One corresponds to the known attributes of a node. For example, we can use the users' social network profile to characterize his node. The second type of feature comes from the set of connections between the nodes of the graph. For example, we can connect users if they are friends in a social network. This is what allows graph-community detection algorithms to find communities based on the network structure, which contrasts with clustering approaches where important relationships are not considered since only the node attributes are modeled.

In the graph-community detection framework, it is possible to explore different properties of the graph to find relevant communities. Below, we summarize the approach of the different algorithms that will be used in the experiments:

Label Propagation (LP) (Raghavan et al., 2007): starts with each node assigned to a different community (label). At each iteration, nodes are assigned with the most frequent label among its neighbors. Ties are broken uniformly and randomly. The algorithm terminates when there are no changes in the labels.

Bigclam (Yang and Leskovec, 2013): optimizes the likelihood community membership metric.

Clauset-Newman-Moore (CNM) (Clauset et al., 2004a): based on the *modularity* criterion. High modularity nodes have dense intra-community connections and sparse inter-community connections. The algorithm evaluates the modularity when removing a node from its community and placing it in its neighbors. If the modularity increases, the node is reassigned. The process stops if there are no reassignments.

Louvain (Blondel et al., 2008): similar to the previous algorithm, with the difference that, at the end of each iteration, builds a new graph by merging all nodes in the same community.

Leading Eigenvector (Newman, 2006): the algorithm starts by having all nodes belonging to the same community. In each iteration, the graph is split into the two communities that increase the modularity the most. The split is determined by evaluating the leading eigenvector of the modularity matrix.

Fast Greedy (Clauset et al., 2004b): the algorithm starts with each node assigned to a different community. The algorithm works by merging nodes in single element communities with other communities if the modularity of the graph increases.

Walktraps (Pons, 2006): the algorithm is based on random walks. A random walk means that we start at a node, we pick a neighbor at random and move to it, then repeat the procedure. By repeating this procedure, it is possible to compute statistics about the visited nodes. The statistics are summarized in a transition matrix, which expresses the probability of going from one node to another through a random walk of length t. Using the distance metric, graph-community detection is then approached as a clustering task.

An effective graph-community detection method to find a structured organization of documents is described in Shahaf et al. (2012). We now propose a similar approach to the task of topic identification across segments from different documents. This allows to study how a pipeline strategy compares to jointly modeling segmentation and topic identification. The graph-community detection problem in the context of topic identification is formalized as follows:

Input: a weighted co-occurrence graph $G_{co} = (W, E)$, where W is the set of nodes and E the set of edges. W corresponds to the set of words from a given set S of document segments. An edge (w_i, w_j) exists if words w_i and w_j occur in some segment $S_i \in S$.

Output: a mapping from each word $w_i \in W$ to a particular community $c \in 1, ..., C$.

Appropriately setting the weights $w(i, j)$ of the edge is a topic of research in the proposed topic identification framework. Depending on how these weights are set, different word communities can be obtained. Therefore, it is necessary to develop an appropriate weighting scheme for the cross-document topic relationship identification task. We hypothesize that having all word co-occurrence contribute in the same way to a weighted score might not be suitable. Therefore, we propose the following *tf-idf*-based weighting schemes:

- **Count**: the number of times the words co-occurred in different segments (equivalent to Shahaf et al. (2012)).
- **Best *tf-idf***: the sum of the highest *tf-idf* values of the words.
- **Count + Best *tf-idf***: the sum of the previous weights.
- **Count + Avg *tf-idf***: the sum of the *count* weight and the sum of the average *tf-idf* values of the words.

After obtaining the word-communities, we need to map them to segments. Segments that get mapped to the same word community are considered to have the same topic. This is done based on a scoring function between a segment and a community. More formally:

$$\underset{c \in C}{argmax}\ score(seg, c), \tag{4.25}$$

where C is the set of communities in G_{co}, and seg is the set of words in a segment. Different formulations of the $score$ function can be designed. We considered the following scoring functions:

$$score_c(seg, c) = \frac{|seg \cap c|}{|c|}, \tag{4.26}$$

$$score_{seg}(seg, c) = \frac{|seg \cap c|}{|seg|}, \tag{4.27}$$

$$score_{tfif}(seg, c) = \frac{\sum\limits_{w_i}^{|seg \cap c|} tfidf(w_i)}{\sum\limits_{w_i}^{seg} tfidf(w_i)} \tag{4.28}$$

The first two functions count the common words between the segment and the community. The score is normalized either by the total number of words in c or seg. The previous functions treat all words in the same way. Therefore, we also define a function that makes words contribute according to their relevance, $score_{tfidf}$. The difference is that common words have a score corresponding to their normalized *tf-idf* value.

Our proposed approach is generic since it is not tied to any particular community detection algorithm. Therefore, we survey multiple algorithms to find the most suitable one for the topic identification task.

4.4 BeamSeg Segmentation

In the following sections, we describe our main proposed model, BeamSeg, a Bayesian unsupervised generative model to address the tasks of breaking documents into incoherent segments and identifying similar topics. The previous model extensions only deal with multi-document segmentation. We now assume that both tasks are related, and, thus, there are advantages in modeling both problems jointly. Moreover, the model leverages segmentation and topic identification by assuming that segment vocabulary usage relationships exist and that segment length properties should modeled at the document modality level[4].

[4] A summary of the notation description used in the mathematical expressions can be found in Appendix A, Table A.3

4.4.1 Bayesian Model for Multi-Document Segmentation

In BeamSeg, all u utterances with a topic k have their bag-of-words representation $\mathbf{x}_u$ drawn from language model ϕ_{z_u}; where z_u is the hidden topic assignment of u. This is in the same spirit as topic modeling approaches such as LDA (Blei et al., 2003), but here the inherent topics are constrained to yield linear segmentations by having topics occurring at most once per document. This constraint induces higher likelihood segmentations to have language models concentrating probability mass on a small subset of the vocabulary. Conversely, low likelihood segmentations spread the probability mass on a broader set of words. This modeling behavior is akin to the lexical cohesion theory. Multi-document segmentation emerges by assuming that topics can be shared across documents. The advantage of this perspective is that we can use segment length properties of similar documents (Section 4.4.2) and better estimate language models with samples from all documents (Section 4.4.3). To model interactions between lexical distributions, we use a dynamic prior which assumes that the mean word probabilities change smoothly across topics (Section 4.4.4).

During inference, we want to find the hidden language models Φ and the topic assignments $\mathbf{z}$ that maximize the likelihood of the joint distribution of the model. Since we only care about segmentation, this process can be simplified by marginalizing out Φ (Section 4.4.3). This enables search to be carried out only in the segmentation space. The linear segmentation constraint has been used to make inference tractable by exhaustively exploring the segmentation space to obtain the exact MAP estimation (Eisenstein and Barzilay, 2008). Therefore, inference amounts to finding the segmentation $\hat{\mathbf{z}} = \operatorname{argmax}_{\mathbf{z}} p(\mathbf{X}|\mathbf{z})p(\mathbf{z})$. We also follow this MAP estimation approach to inference, but given a multi-document setting, this approach is not feasible, as the segments can share topics. We address this by using a beam search algorithm during inference, which allows the segmentation procedure to recover from early mistakes (Section 4.4.5).

4.4.2 Segment Length Prior

The $\hat{\mathbf{z}} = \operatorname{argmax}_{\mathbf{z}} p(\mathbf{X}|\mathbf{z})p(\mathbf{z})$ expression we want to maximize to obtain the most likely segmentation puts a prior, $p(\mathbf{z})$, on the segment length of the target documents. Given the approach of searching the segmentation space only during inference, we can plug in different segment length priors to see how they behave during the segmentation task. One of such distribution is the Beta-Bernoulli, which has been used before in the PLDA model (Purver et al., 2006) and also in our PLDA-MD extension (Section 4.2.2):

$$p(\mathbf{z}) = \left(\frac{\Gamma(2\gamma)}{\Gamma(\gamma)^2}\right)^D \prod_{d=1}^{D} \frac{\Gamma(n_1^d + \gamma)\Gamma(n_0^d + \gamma)}{\Gamma(U_d + 2\gamma)}, \tag{4.29}$$

We also propose a Gamma-Poisson distributed segment length prior. In this setup, we assume that the document topic shift probabilities π are drawn from a Gamma prior parameterized by α and β:

$$\begin{aligned} p(\pi|\alpha,\beta) &= \prod_{d=1}^{D} \frac{\beta^{\alpha}}{\Gamma(\alpha)} \pi_d^{\alpha-1} e^{-\beta\pi_d} \\ &= \left(\frac{\beta^{\alpha}}{\Gamma(\alpha)}\right)^{D} \prod_{d=1}^{D} \pi_d^{\alpha-1} e^{-\beta\pi_d} \end{aligned} \tag{4.30}$$

$p(\mathbf{z}|\pi)$ is the probability of the utterances being a segment boundary given the topic shift probabilities. These are Poisson distributed and defined as follows:

$$p(\mathbf{z}|\pi) = \prod_{d=1}^{D} \pi^{n_1^d} e^{-U_d\pi} \tag{4.31}$$

Putting the two equations together yields:

$$\begin{aligned} p(\mathbf{z}|\pi)p(\pi|\alpha,\beta) &= \left(\frac{\beta^{\alpha}}{\Gamma(\alpha)}\right)^{D} \prod_{d=1}^{D} \pi^{n_1^d} e^{-U_d\pi} \pi_d^{\alpha-1} e^{-\beta\pi_d} \\ &= \left(\frac{\beta^{\alpha}}{\Gamma(\alpha)}\right)^{D} \prod_{d=1}^{D} \pi^{n_1^d+\alpha-1} e^{-(U_d+\beta)\pi} d\pi \end{aligned} \tag{4.32}$$

Noting that Equation 4.32 is also a Gamma distribution, we can see that the Gamma and Poisson distributions are conjugate. Therefore, can we use a marginalization process where the parameters are integrated out as follows:

$$\begin{aligned} p(\mathbf{z}) &= \int p(\mathbf{z}|\pi)p(\pi|\alpha,\beta)d\pi \\ &= \left(\frac{\beta^{\alpha}}{\Gamma(\alpha)}\right)^{D} \prod_{d=1}^{D} \int \pi^{n_1^d+\alpha-1} e^{-(U_d+\beta)\pi} d\pi_d \\ &= \left(\frac{\beta^{\alpha}}{\Gamma(\alpha)}\right)^{D} \prod_{d=1}^{D} \frac{\Gamma(n_1^d+\alpha)}{(U_d+\beta)^{n_1^d+\alpha}} \int \frac{(U_d+\beta)^{n_1^d+\alpha}}{\Gamma(n_1^d+\alpha)} \pi^{n_1^d+\alpha-1} e^{-(U_d+\beta)\pi} d\pi_d \\ &= \left(\frac{\beta^{\alpha}}{\Gamma(\alpha)}\right)^{D} \prod_{d=1}^{D} \frac{\Gamma(n_1^d+\alpha)}{(U_d+\beta)^{n_1^d+\alpha}} \end{aligned} \tag{4.33}$$

Applying the priors based on the document modality can be done if we the modality is known *a priori*, which is the approach we take. It is only necessary to have individual hyperparameters for each modality and apply them according to the document for which we are computing the segmentation likelihood.

4.4.3 Language Models

Using the previous setup, we define the joint likelihood as follows:

$$\begin{aligned} p(\mathbf{X}|\mathbf{z},\Phi) &= \prod_{u}^{U} p(\mathbf{x}_u|\phi_{z_u}) \prod_{k}^{K} p(\phi_k|\beta) \\ &= \prod_{k}^{K} p(\phi_k|\beta) \prod_{\{u:z_u=k\}} p(\mathbf{x}_u|\phi_k) \\ &= \Big(\frac{\Gamma(W\beta)}{\Gamma(\beta)^W}\Big)^K \prod_{k=1}^{K}\prod_{w=1}^{W} \phi_{k,w}^{\beta-1}\phi_{k,w}^{n_{U,w}^k} \\ &= \Big(\frac{\Gamma(W\beta)}{\Gamma(\beta)^W}\Big)^K \prod_{k=1}^{K}\prod_{w=1}^{W} \phi_{k,w}^{n_{U,w}^k+\beta-1}, \end{aligned} \tag{4.34}$$

where $\mathbf{X}$ is the set of all U utterances in the document collection; K is the number of topics; and β are the Dirichlet prior parameters from which Φ is drawn.

Since we only care about segmentation and topic identification, inference can be simplified by analytically marginalizing out the hidden language models Φ. This enables search to be carried out only in the segmentation space. The marginalization process is performed by appealing to the conjugacy between Categorical language models and the Dirichlet prior. This allows the conjugate Dirichlet distribution to integrate to one, leaving the marginalized joint likelihood expression with the normalizing constants:

$$\begin{aligned} p(\mathbf{X}|\mathbf{z}) &= \int p(\mathbf{X}|\mathbf{z},\Phi)p(\Phi|\beta)d\Phi \\ &= \int \Big(\frac{\Gamma(W\beta)}{\Gamma(\beta)^W}\Big)^K \prod_{k=1}^{K}\prod_{w=1}^{W} \phi_{k,w}^{n_{U,w}^k+\beta-1} d\phi_k \\ &= \Big(\frac{\Gamma(W\beta)}{\Gamma(\beta)^W}\Big)^K \prod_{k=1}^{K} \frac{\prod_{w=1}^{W}\Gamma(n_{U,w}^k+\beta)}{\Gamma(n_U^k+W\beta)} \int \frac{\Gamma(n_U^k+W\beta)}{\prod_{w=1}^{W}\Gamma(n_{U,w}^k+\beta)} \prod_{w=1}^{W} \phi_{k,w}^{n_{U,w}^k+\beta-1} d\phi_k \\ &= \Big(\frac{\Gamma(W\beta)}{\Gamma(\beta)^W}\Big)^K \prod_{k=1}^{K} \frac{\prod_{w=1}^{W}\Gamma(n_{U,w}^k+\beta)}{\Gamma(n_U^k+W\beta)}, \end{aligned} \tag{4.35}$$

where W is the vocabulary set; $n_{U,w}^k$ is number of times word w is assigned topic k in all U utterances of the document collection; n_U^k is number of times topic k appears in U; and the symbol Γ refers to the Gamma function. The resulting expression in Equation 4.35 corresponds to the product of the individual topic likelihoods, comprised of segments from different documents.

4.4.4 Dynamic Language Model Prior

The previously described prior assumes that language model draws are exchangeable. Therefore, they are independent of each other and cannot encode dynamic relationship between them. In a segmentation scenario where documents follow an overarching subject, this may not be a reasonable assumption. We hypothesize that, in these cases, language models change smoothly across topics by establishing a dynamic between the previous and the current prior parameters. This modeling of topics as a time series can be found in other works (Blei and Lafferty, 2006a,b; He et al., 2017; Huang, 2018; Jahnichen et al., 2018). In BeamSeg, we adopt a perspective similar to topic tracking (Watanabe et al., 2011) to model such topic interactions. Similarly to topic tracking, we factor the β prior parameters in $\alpha_k \hat{\phi}_{k'}$, a precision and mean word probabilities parameters. Assuming some fixedtopic order, k indexes the parameters of a topic, and k' the parameters of the previous one. The α_k precision represents the persistence of word usage throughout topics. The $\hat{\phi}_k$ parameters model the language model dynamics by assuming that the mean word probabilities at k are the same as those at k'. Our approach entails that a single chain of language models is used, affording multi-document segmentation. This contrasts with the multiple chains in the original topic tracking model. Another difference is that topic tracking assumes a given segmentation based on a fixed-length window. This is because the goal of the model is to find good language models and not segment documents.

To compute the likelihood of the joint under this prior it is necessary to determine the parameters for all $k \in K$. This is a two-fold process, where we first update the α_k precision parameter using the expression derived from Minka (2000):

$$\alpha_k = \alpha_k \frac{\sum\limits_{w=1}^{W} \hat{\phi}_{k'w}(\Psi(n_{U,w}^k + \alpha_k \hat{\phi}_{k'w}) - \Psi(\alpha_k \hat{\phi}_{k'w}))}{\Psi(n_U^k + \alpha_k) - \Psi(\alpha_k)}, \tag{4.36}$$

where Ψ is the digamma function. Then, we update the mean word probability parameters:

$$\hat{\phi}_{kw} = \frac{n_{U,w}^k + \alpha_k \hat{\phi}_{k'w}}{n_U^k + \alpha_k} \tag{4.37}$$

The update equations are sequentially applied according to the considered topic ordering. By following this process we can deal with long-range dependencies by taking into account the data contribution at each k. Finally, we plug-in the obtained prior parameters in the join likelihood formula in Equation 4.35:

$$p(\mathbf{X}|\mathbf{z}) = \prod_{k=1}^{K} \frac{\Gamma(\sum_{w=1}^{W} \alpha_k \hat{\phi}_{kw})}{\prod_{w=1}^{W} \Gamma(\alpha_k \hat{\phi}_{kw})} \prod_{k=1}^{K} \frac{\prod_{w=1}^{W} \Gamma(n_{U,w}^k + \alpha_k \hat{\phi}_{kw})}{\Gamma(n_U^k + \sum_{w=1}^{W} \alpha_k \hat{\phi}_{kw})}, \tag{4.38}$$

The presented language model prior assumes that a single topic ordering exists. Consequently, ideally all document would have the same topic ordering, but in practice they do not. However, we still expect that a significant fraction of the topic order is shared in different documents. Whether this simplification still allows to improve segmentation and topic identification is a subject we study in the experiments.

4.4.5 Inference

Having specified the model, we now turn to the inference problem. We study two different approaches to inference: MAP and VI. Another alternative for inference would be Gibbs sampling. However, the difficulty in applying it to our model is in its slow convergence to the stationary distribution, due to the tight coupling of the hidden variables induced by the linear segmentation constraint in a multi-document scenario. In the next sections, we demonstrate that the MAP approach affords an approximation that can be implemented in practice and that VI is not suitable for a multi-document segmentation and topic identification scenario.

4.4.5.1 Maximum a Posteriori and Beam Search

During inference, we want to find the hidden set of language models Φ and the topic vector assignment $\mathbf{z}$ that maximize the likelihood of the joint distribution of the model. Since the language models were marginalized out, inference amounts to finding the segmentation $\hat{\mathbf{z}} = \text{argmax}_{\mathbf{z}}\ p(\mathbf{X}|\mathbf{z})p(\mathbf{z})$. Using the marginalized joint likelihood, an approximation of $\hat{\mathbf{z}}$ can be obtained using a beam search algorithm.

Following the approach in BayeSeg (Eisenstein and Barzilay, 2008), inference is carried out as an optimization problem, where the target segmentation maximizes the objective function defined by the joint likelihood in Equation 4.35. Contrary to the single-document BayeSeg model, we assume that language models aggregate segments from different documents, making an exhaustive exploration of the segmentation space intractable. We address this problem by combining beam search and a greedy segmentation procedure. We define $\mathbf{z}_j^*$ as the segmentation that maximizes the objective function up to utterance j. Considering the topic assignment $z_j = k$ and the previous segmentation $\mathbf{z}_{j-1}$, the value for the objective function is written,

$$s(k, j, \mathbf{z}_{j-1}) = p(\{\mathbf{x}_0...\mathbf{x}_j\}|\mathbf{z}_{j-1}, z_j = k) \tag{4.39}$$

Using a recursive definition, we obtain the optimal segmentation using:

$$\mathbf{z}_j^* = \underset{k \in K}{\text{argmax}}\, s(k, j, \mathbf{z}_{j-1}^*) \tag{4.40}$$

This is a greedy sampling approach since it makes incremental decisions when finding the highest likelihood segmentation. This is error-prone since we might need to take into account subsequent utterances to discover higher likelihood configurations. Moreover, once a mistake is made there is no way to recover. To address this problem, we add beam search to the algorithm by keeping track of all topic assignments, instead of just the highest likelihood one. At the end of each recursive step, we prune the top-b segmentations. To run the greedy sampler, we iteratively slide a window of length l in each document, following the utterance order of the documents. By using this process, we are able to take into account utterances from the whole collection before reaching the end of a document, which enables multi-document topic segments to emerge.

4.4.5.2 Variational Inference and Beam Search

In this section, we propose an alternative inference method. The goal is to approximate the posterior distribution of the model instead of finding the parameters that maximize the joint. The advantage is that we can use as much information as possible from observed data to better estimate parameters. To this end, we use Variational Inference (VI) with a mean-field approach. The setup uses the same marginalization process of the latent variables as before, and, thus, we use collapsed VI (Teh et al., 2007), which leverage insights from collapsed Gibbs sampling. For example, in LDA, working in a collapsed space affords better mixing times when compared to a Gibbs sampler that samples all latent variables. This suggests that there is a coupling between the variables and the parameters. By marginalizing out the parameters, new dependencies between latent variables are introduced, but these are spread out over many latent variables. The implication is that the dependency between any two latent variables is expected to be small. This is a scenario suitable for mean-field VI: a particular variable interacts with the remaining variables only through the chosen family of variational distributions, making the impact of any single variable very small (Teh et al., 2007).

The attractive theoretical advantages and the fact that this VI setup has been successfully used for segmentation (Eisenstein, 2009) motivated us to pursue a similar approach in BeamSeg. As we will detail later, it turns out that this VI setup is not suitable for our model. Nonetheless, we provide the derivations that led to this conclusion. One note before moving on the derivations: to simplify the notation we will use the β prior, but the dynamic prior described in Section 4.4.4 can be added by substituting β with $\alpha_k \hat{\phi}_{k'w}$.

We now describe the mean-field VI setup for the latent variables $\mathbf{z}$, the only ones remaining after marginalization. First, we define the variational distribution family $q(\mathbf{z}) \in \mathcal{Q}$:

$$q(\mathbf{z}) = \prod_{d=1}^{D} \prod_{i=1}^{d} q(z_{d,i}|\gamma_{d,i}), \tag{4.41}$$

where the topic assignment $q(z_{d,i}|\gamma_{d,i})$ has Categorical parameters $\gamma_{d,i}$ with dimension K. It should be noted that we are placing variational parameters on each word i in document d, whereas before we were directly plugging in the word counts from a full utterance. This a consequence of estimating the full posterior, where we want to know what is the probability of the words belonging to each of the topics. The variational distribution $q(\mathbf{z})$ is fully factorized, and, thus, all variational parameters are assumed to be independent of each other. For optimizing the ELBO we use the CAVI approach, previously introduced in Section 4.1.3. The goal of this approach is to obtain the variational parameters update equations and iteratively apply them until convergence. The first step consists in rewriting the ELBO using iterated expectation and absorbing into a constant the terms that do not depend on the variational factor $q_{d,i}(z_{d,i})$:

$$\text{ELBO}(q_{d,i}(z_{d,i})) = \mathbb{E}_{d,i}[\mathbb{E}_{\neg d,i}[\log p(z_{d,i}, \mathbf{z}_{\neg d,i}, \mathbf{x})]] - \mathbb{E}_{d,i}[\log q_{d,i}(z_{d,i})] + const. \tag{4.42}$$

Then, we take the gradient with respect to individual variational parameters $\gamma_{d,i}^k$, which results in:

$$\gamma_{d,i}^k = q_{d,i}(z_{d,i} = k) = \frac{\exp(\mathbb{E}_{q(\mathbf{z}_{\neg d,i})}[\log p(z_{d,i} = k|\mathbf{x}, \mathbf{z}_{\neg d,i}, \beta)])}{\sum_{k'}^{K} \exp(\mathbb{E}_{q(\mathbf{z}_{\neg d,i})}[\log p(z_{d,i} = k'|\mathbf{x}, \mathbf{z}_{\neg d,i}, \beta)])} \tag{4.43}$$

We now work on the $\log p(z_{d,i} = k|\mathbf{x}, \mathbf{z}_{\neg d,i}, \beta)$ expression from Equation 4.43. The expression is simplified because the numerator and the denominator only differ on the $z_{d,i}$ variable, which is similar to the approach in the derivations for the Gibbs sampler in PLDA-MD. The derivation starts by using the conditional probability rule, revealing the expression for the joint probability distribution of the model. The complete derivations for the joint have been presented before (Section 4.4.3), and, thus, we gloss over its details now. The rest of the derivation is presented below:

$$\begin{aligned}
\log(p(z_{d,i} = k|\mathbf{z}_{\neg d,i}, \mathbf{x}, \beta)) &= \log\Big(\frac{p(z_{d,i}^k, \mathbf{z}_{\neg d,i}, \mathbf{x}, \beta)}{p(\mathbf{z}_{\neg d,i}, \mathbf{x}, \beta)}\Big) \\
&= \log\left(\frac{\prod_{k'}^{K} C \frac{\prod_{w=1}^{W}\Gamma(n_w^{D_k}+\beta)}{\Gamma(\sum_{w=1}^{W} n_w^{D_k}+\beta)}}{\prod_{k'}^{K} C \frac{\prod_{w=1}^{W}\Gamma(n_w^{D_{k,\neg d,i}}+\beta)}{\Gamma(\sum_{w=1}^{W} n_w^{D_{k,\neg d,i}}+\beta)}}\right) \\
&= \log\left(\frac{\frac{\prod_{w=1}^{W}\Gamma(n_w^{D_k}+\beta)}{\Gamma(\sum_{w=1}^{W} n_w^{D_k+\beta})}}{\frac{\prod_{w=1}^{W}\Gamma(n_w^{D_{k,\neg d,i}}+\beta)}{\Gamma(\sum_{w=1}^{W} n_w^{D_{k,\neg d,i}}+\beta)}}\right) \\
&= \log\left(\frac{\prod_{w=1}^{W}\Gamma(n_w^{D_k}+\beta)}{\prod_{w=1}^{W}\Gamma(n_w^{D_{k,\neg d,i}}+\beta)} \frac{\Gamma(\sum_{w=1}^{W} n_w^{D_{k,\neg d,i}}+\beta)}{\Gamma(\sum_{w=1}^{W} n_w^{D_k}+\beta)}\right),
\end{aligned} \tag{4.44}$$

where $n_w^{D_k}$ is the number of k topic assignments to word w in dataset D, and $n_w^{D_{k,\neg d,i}}$ is similar but excludes the i^{th} word of d from the counts. We simplify the first factor in Equation 4.44 using $\Gamma(n) = (n-1)\Gamma(n-1)$:

$$\begin{aligned}\frac{\prod_{w=1}^{W}\Gamma(n_w^{D_k}+\beta)}{\prod_{w=1}^{W}\Gamma(n_w^{D_{k,\neg d,i}}+\beta)} &= \frac{\Gamma(n_{w_{d,i}}^{D_k}+\beta)}{\Gamma(n_{w_{d,i}}^{D_{k,\neg d,i}}+\beta)}\\ &= \frac{\Gamma(n_{w_{d,i}}^{D_k}+\beta)}{\Gamma(n_{w_{d,i}}^{D_k}+\beta-1)}\\ &= \frac{\Gamma(n_{w_{d,i}}^{D_k}+\beta-1)(n_{w_{d,i}}^{D_k}+\beta-1)}{\Gamma(n_{w_{d,i}}^{D_k}+\beta-1)}\\ &= n_{w_{d,i}}^{D_k}+\beta-1 \end{aligned} \tag{4.45}$$

A similar process is carried out for the second factor of Equation 4.44:

$$\begin{aligned}\frac{\Gamma(\sum_{w=1}^{W} n_w^{D_{k,\neg d,i}}+\beta)}{\Gamma(\sum_{w=1}^{W} n_w^{D_k}+\beta)} &= \frac{\Gamma((\sum_{w=1}^{W} n_w^{D_k}+\beta)-1)}{\Gamma((\sum_{w=1}^{W} n_w^{D_k}+\beta)-1)((\sum_{w=1}^{W} n_w^{D_k}+\beta)-1)}\\ &= \frac{1}{(\sum_{w=1}^{W} n_w^{D_k}+\beta)-1}\\ &= \frac{1}{n_\cdot^{D_k}+W\beta-1}, \end{aligned} \tag{4.46}$$

where $n_\cdot^{D_k}$ is total number of times topic k appears in D. Plugging in the factors back yields:

$$\begin{aligned}\log(p(z_{d,i}=k|\mathbf{z}_{\neg d,i},\mathbf{x},\beta)) &= \log\left(\frac{n_{w_{d,i}}^{D_k}+\beta-1}{n_\cdot^{D_k}+W\beta-1}\right)\\ &= \log(n_{w_{d,i}}^{D_k}+\beta-1)-\log(n_\cdot^{D_k}+W\beta-1) \end{aligned} \tag{4.47}$$

The final expression for the variational parameters update is then,

$$\gamma_{d,i}^{k} = \frac{\exp(\mathbb{E}_{q(z_{\neg d,i})})[\log(n_{w_{d,i}}^{D_k}+\beta-1)-\log(n_\cdot^{D_k}+W\beta-1)]}{\exp(\sum_{k'=1}^{K}\mathbb{E}_{q(z_{\neg d,i})})[\log(n_{w_{d,i}}^{D_{k'}}+\beta-1)-\log(n_\cdot^{D_{k'}}+W\beta-1]} \tag{4.48}$$

Computing the exact expectations in Equation 4.48 is computationally expensive. In the collapsed VI framework this problem is addressed using a Gaussian approximation, which has a much lower computational cost. Assuming that $n_{w_{d,i}}^{D} \gg 0$, and noting that $n_{w_{d,i}}^{D_{k,\neg d,i}} = \sum_{d=1}^{D}\sum_{i'=0,i'\neq i}^{d}\mathbb{1}(z_{d,i'}=k)$ is a sum of a large number independent Bernoulli variables $\mathbb{1}(z_{d,i'}=k)$ with mean parameter $\gamma_{d,i'}^{k}$, an accurate approximation by a Gaussian can be made. Under these assumptions, the mean and the variance are given by the sum of the means and the variances of the individual Bernoulli variables:

$$n_{w_{d,i}}^{D_k} + \beta - 1 = n_{w_{d,i}}^{D_k,\neg d,i} + \beta \tag{4.49}$$

$$\mathbb{E}_q[n_{w_{d,i}}^{D_k,\neg d,i}] = \sum_{d=1}^{D} \sum_{i'=1,i'\neq i}^{d} \gamma_{d,i'}^{k} \delta(w_{d,i}, w_{d,i'}) \tag{4.50}$$

$$\mathrm{Var}_q[n_{w_{d,i}}^{D_k,\neg d,i}] = \sum_{d=1}^{D} \sum_{i'=1,i'\neq i}^{d} (1 - \gamma_{d,i'}^{k}) \delta(w_{d,i}, w_{d,i'}) \tag{4.51}$$

Using the previous derivation, we can approximate $\mathbb{E}_q[\log(n_{w_{d,i}}^{D_k,\neg d,i} + \beta)]$ factor from Equation 4.48 with a second-order Taylor expansion:

$$\mathbb{E}_q[\log(n_{w_{d,i}}^{D_k,\neg d,i} + \beta)] \approx \log(\beta + \mathbb{E}_q[n_{w_{d,i}}^{D_k,\neg d,i}]) - \frac{\mathrm{Var}_q[n_{w_{d,i}}^{D_k,\neg d,i}]}{2(\beta_w + \mathbb{E}_q[n_{w_{d,i}}^{D_k,\neg d,i}])^2} \tag{4.52}$$

A similar setting is found for $n_{\cdot}^{D_k}$, and, thus, a Gaussian approximation can also be made:

$$\mathbb{E}[n_{\cdot}^{U'_k}] = \sum_{d=1}^{D} \sum_{i'=1,i'\neq i}^{d} \gamma_{d,i'}^{k'} \tag{4.53}$$

$$\mathrm{Var}[n_{\cdot}^{U'_k}] = \sum_{d=1}^{D} \sum_{i'=1,i'\neq i}^{d} \gamma_{d,i'}^{k'} (1 - \gamma_{d,i'}^{k'}) \tag{4.54}$$

Then, using again a second-order Taylor expansion, we obtain for $\log(n_{\cdot}^{D_k} + W\beta)$:

$$\mathbb{E}_q[\log(n_{\cdot}^{D_k} + W\beta)] \approx \log(W\beta + \mathbb{E}[n_{\cdot}^{D_k}]) - \frac{\mathrm{Var}[n_{\cdot}^{D_k}]}{2(W\beta + \mathbb{E}[n_{\cdot}^{D_k}])^2} \tag{4.55}$$

Plugging in the approximation from Equations 4.52 and 4.53 back to variational parameters update Equation 4.48 yields the final expression:

$$\gamma_{d,i}^{k} \propto (\beta_w + \mathbb{E}_q[n_{w_{d,i}}^{D_k,\neg d,i}])(W\beta + \mathbb{E}[n_{\cdot}^{D_k}])^{-1} \exp\left(- \frac{\mathrm{Var}_q[n_{w_{d,i}}^{D_k,\neg d,i}]}{2(\beta_w + \mathbb{E}_q[n_{w_{d,i}}^{D_k,\neg d,i}])^2} + \frac{\mathrm{Var}[n_{\cdot}^{D_k}]}{2(W\beta + \mathbb{E}[n_{\cdot}^{D_k}])^2} \right) \tag{4.56}$$

Notice that in the resulting update expression derived above $z_{d,i}^{k}$ only depends on $\mathbf{z}_{\neg d,i}^{k}$ through the $n_{w_{d,i}}^{D_k,\neg d,i}$ and $n_{\cdot}^{D_k}$ counts. This makes the case for arguing that the variables are only weakly dependent on each other. In such settings, variational inference is expected to be accurate (Teh et al., 2007), since we can safely replace the latent variables by independent variational parameters.

The final procedure for multi-document segmentation and topic identification iterates between segmenting documents based on the current state of the variational parameters and then reestimate them based on the obtained segmentation. The overall blueprint of the procedure is similar to the hierarchical segmentation procedure in Eisenstein (2009), although important differences exist. To segment the documents we use the dynamic programming approach described in Section 4.2.1. Since we are in a VI setup, the expression for the joint probability takes into account the variational parameters:

$$p(\mathbf{X}|\mathbf{z},\gamma) = \left(\frac{\Gamma(W\beta)}{\Gamma(\beta)^W}\right)^K \prod_{k=1}^{K} \frac{\prod_{w=1}^{W} \Gamma(n_w^{D_k}\gamma_w^{D_k} + \beta)}{\Gamma(n_\cdot^{D_k}\gamma_\cdot^{D_k} + W\beta)}, \tag{4.57}$$

where $\gamma_w^{D_k}$ is the sum of all k^{th} components of the $\gamma_{d,i}^k$ variational parameters for all words that match w. Essentially what happens in the previous rewriting of the joint is that we are scaling the word counts using $\gamma_{d,i}^k$ as weights.

4.4.5.3 Why VI is unsuitable for Multi-Document Segmentation and Topic Identification

The problem with this approach is that it is not possible to compute different likelihoods for different segmentations since we would always add the same scaled words counts to different topics. This contrasts with the hierarchical segmentation scenario because the variational parameters model the assignments of words to the different levels of the hierarchy. Thus, when computing the likelihood of the hierarchical segmentation, the segments counts are scaled according to the variational parameters, yielding different likelihoods for different segmentations. In our case, to observe different likelihood estimations we need to commit to a segmentation at each iteration of the dynamic programming algorithm. The segmentation likelihood in each cell of the matrix is then the sum of committed segmentations up to that utterance, plus the scaled likelihood of the segment currently being considered. After computing segmentation likelihoods for a line of the matrix, the cell with the highest value will provide a new segmentation to be committed. This brings up another problem, which is the fact that we do not have a straightforward way of assigning segments to topics since we only have the value of the variational parameters of each word. Ultimately, all these difficulties led us to believe that this VI setup is not suitable for a multi-document segmentation scenario. We still did some experiments with a procedure that assigns segments to the topic with the highest sum of variational parameters values. This type of solution is not suitable since few words with high variational parameters values bias the inference procedure to the corresponding topic. Furthermore, these high values do not change much with the variational updates during the dynamic programming procedure. This is problematic because the iterations are computationally expensive. More standard VI setups are efficient because they only require the variational update step. In our case, we need to find the best segmentation, which is done by using a dynamic programing procedure. This overhead makes this two-step procedure hard to scale. Under these conditions, the segmentation quality was low, and we abandoned this research line.

5 Segmentation Evaluation

Having collected a suitable dataset (Chapter 3) and proposed multi-document segmentation models (Chapter 4), we can now compare the performance of single and multi-document (joint and non-joint) segmentation models. This evaluation also allows to test our hypothesis that modeling segment vocabulary usage relationships and segment length characteristics at the document level can improve segmentation. In this context, we perform three segmentation-related experiments. For these experiments, we define the evaluation metrics in Section 5.1. In the first experiment, we investigate how using a caching scheme in a Gibbs sampler in PLDA-MD influences the segmentation results in a synthetic dataset (Section 5.2). Then, we analyze how existing single and multi-document models perform in our learning materials dataset. Finally, we evaluate the performance of the proposed BeamSeg algorithm according to different prior assumptions based on language model independence and document modality (Section 5.4).

5.1 Evaluation Metrics

To evaluate segmentation, we considered the standard evaluation metrics for this task, which have been discussed and described in Section 3.3.2. We report the WD score since it is a widely used metric in the literature and deals with the drawbacks of the P_k metric. For consistency, we take the output segmentations from all systems and evaluate them using the same software (the python module `segeval` (Fournier, 2013)). For the window size, we use the recommended average segment length of the reference.

5.2 PLDA-MD Gibbs Sampling

In Section 4.2.2, we proposed a caching scheme to speed up the computation of the Gibbs sampling equations. The caching scheme requires a systematic (fix) scan order to sample the variables. This raises the question if such restriction affects the convergence of the Gibbs sampler. Therefore, in this experiment, we compare the convergence of a Gibbs sampler for PLDA-MD using a random scan order and a systematic scan order. If the convergence is not affected, we can expect that the segmentation is also not affected.

5.2.1 Experimental Setup

To control this experiment as much as possible, such that the only variable interfering is the scan order, we generate a synthetic dataset using PLD-MD generative process. Following this process, we obtain a dataset with 50 documents, 15000 sentences, and 75000 words from a vocabulary with size 100. In order to generate related documents, the same topic proportions were reused across different documents. We fixed the hyperparameters of the model to: $\alpha = 0.7$, $\beta = 0.7$, $\gamma_1 = 0.2$, and $\gamma_2 = 10$[1]. The reason for choosing these α and β values was to obtain topic proportions with multiple topic spikes, instead of having a single topic with high probability, which would make the segmentation task easier. The γ were set such that the majority of sentences would have $c_{d,u} = 0$, while still allowing some variability in the expected segment length. Using these parameters in the generative model a total of 223 segments were obtained. The Gibbs sampler was then run for 10000 iterations with burn-in and lag of 1000. The burn-in period corresponds to the first iterations of the sampler and since its estimates are not accurate they are discarded. The lag value determines the frequency samples are collected (after the burn-in period). The goal is to avoid auto-correlation problems that would occur if we collected samples at every iteration of the sampler. In the end, $c_{d,u}$ variables were considered as segment boundaries if their expected value has higher than a threshold of 0.8. The convergence is assessed by monitoring the log likelihood of the segmentation according to the model at each iteration of the Gibbs sampler. Another metric we monitor in this experiment is WD. It is expected that if the scan order of the variables impact on the Gibbs sampler is small, the obtained segmentations should be similar. To determine if PLDA-MD is an effective multi-document segmentation strategy, we also compare it with its single-document counterpart PLDA.

5.2.2 Experimental Results

Figure 5.1 shows the log likelihood of the current segmentation state throughout the Gibbs sampling iterations. When comparing PLDA-MD and PLDA-MD-c (the caching scheme version for Gibbs sampling), we can see that the segmentation log-likelihood values are close throughout the sampling process. This is evidence that choosing a random or a systematic scan order of the variables does not impact the solution found by the Gibbs sampler. Thus, we can use the caching scheme without sacrificing performance. This is further corroborated by the WD scores from Table 5.1, where the performance of PLDA-MD and PLDA-MD-c is similar. Given the previous close likelihood and WD, we argue that both samplers are converging to

[1] We presented PLDA-MD with a single γ hyperparameter (symmetric prior) for notational convenience but it can be trivially extended for the non-symmetric version using γ_1 and γ_2.

the exact same solution. Despite the slightly better performance of PLDA-MD-c (a 0.01 WD improvement), with more data or more iterations, we expect that both samplers find the same segmentations.

Comparing PLDA-MD-c with PLDA, it is also possible to observe that the latter has a consistently higher segmentation likelihood. This indicates that PLDA-MD-c finds better solutions. Further evidence of this is found when comparing the WD scores. These scores show that PLDA-MD-c has a 4% increase in performance compared to PLDA. In 42 out of 50 tests cases WD performance gains of up to 10% are observed. Only in five test cases, the final segmentation for both models was exactly the same. These results show that it is indeed possible to leverage segmentation on multiple related documents. A closer look at the segmentation results showed that most gains are obtained in smaller segments. This is inline with the difference between the two models. That is, the advantage of PLDA-MD-c is that if a segment is too short to accurately determine its topic portions, the model still takes into account possibly similar segments in other documents. Another result from this experiment is that the cache version of the multi-document model achieves similar WD results. This corroborates that the sampling order of the variables does not impact the segmentation given the same number of Gibbs sampling iterations.

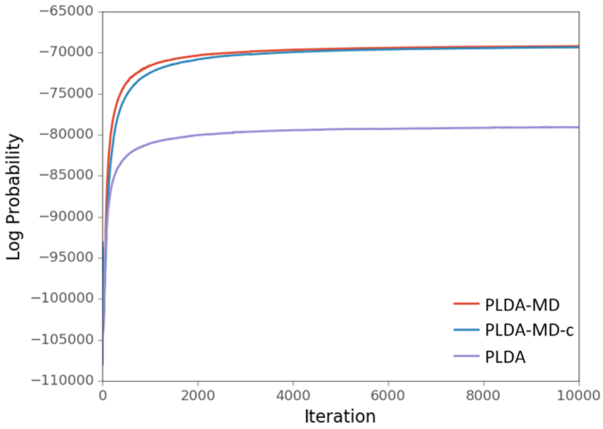


Figure 5.1: Joint probability of the models during Gibbs sampling.

	PLDA	PLDA-MD	PLDA-MD-c
WD	0.33 ± 0.06	0.29 ± 0.06	0.28 ± 0.07

Table 5.1: Average WD scores in the synthetic dataset.

To determine the impact of the caching scheme on the Gibbs sampler's execution time, we plot how long each iteration takes in Figure 5.2. By using cache, the total execution is cut down from 232.2 hours to 110.9 hours, a 52.2% reduction. This makes the Gibbs sampler much more scalable, which is especially relevant in our scenario with a collection of related documents. From the plot, we also gain insight into how the variables are changing during the sampling procedure. We can observe that considerable changes in the state do not often occur since the time to perform each iteration does not vary considerably. This means we can take advantage of caching using the proposed systematic scan order of the variables. There are, however, some exceptions. For example, close to iteration 6000 we can see a period where iterations are taking longer. This is related to the Gibbs sampler finding a better region of the state space to explore. Such periods are not long, and we quickly get back to executing iterations efficiently.

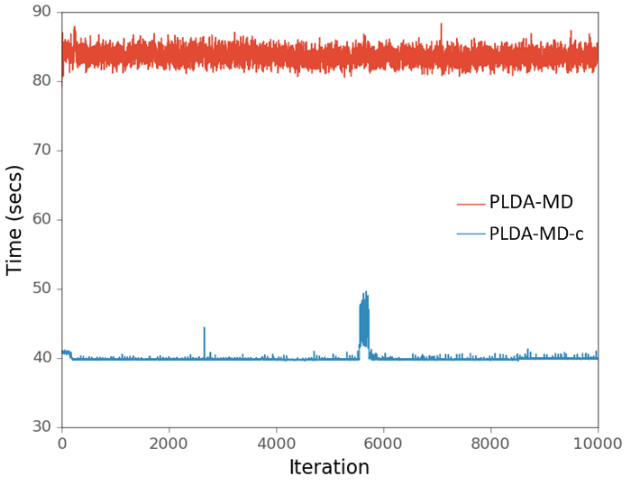


Figure 5.2: Time to perform each Gibbs sampling iteration.

5.3 Benchmark Segmentation

The goal of this experiment is to both determine the baseline results for the BeamSeg proposed approach as well as observing how single and multi-document models perform in a variety of domains.

5.3.1 Experimental Setup

In this experiment, we test several state-of-the-art segmentation algorithms and compare them with our multi-document extensions Bayesseg-MD (Section 4.2.1) and PLDA-MD (Section 4.2.2)). This way we can also study the impact that multi-document models have on the results. The best results from this experiment will serve as a baseline for the proposed BeamSeg model in Section 5.4. In order to cover lexical similarity and probabilistic approaches in single-document model approaches, we tested the following algorithms: Bayesseg, PLDA, CVS, C99, MinCut, and MultiSeg.

As discussed in Section 3.1, the only existing dataset that targets multi-document segmentation is the one provided by Jeong and Titov (2010). We test the algorithms in the Biography, and News domains. We left out the Reports domain due to its fix two segment document structure and the Podcast domain because it only tracks speaker changes. Finally, we also test the algorithms with our learning materials dataset in the AVL trees and Physics domains (Section 3.2), achieving this way a broader domain coverage and testing the algorithms in documents with a strong topic development aspect.

The hyperparameter setup of the models was carried out on a development set. For the Biography, and News domains we picked one of the subjects and all of its documents since each subject has few documents. We did not carry out hyperparameter tuning in these domains for MultiSeg. Instead, we used the configuration that the authors used in their experiments. For the Physics domain, the development set corresponds to a sample of ten documents from one of the subjects. For tuning the hyperparameters to test in the AVL trees domain, we also used a development set from the Physics domain since both datasets are composed of documents with pedagogical content. The Gibbs sampling for PLDA and PLDA-MD run for 20000 iterations with a burn-in period of 1000 iterations and a lag value of 200 iterations. The $c_{d,u}$ utterance variables were considered segment boundaries if a value of 0.8 or higher was obtained. The PLDA-MD model uses the caching feature during Gibbs sampling. The CVS model used the 300 dimensions GloVe (Pennington et al., 2014) word embeddings.

5.3.2 Experimental Results

In the discussion of the experimental results below, we depict the best WD results in bold in each of the corresponding tables.

From the WD averages results in Table 5.2, we can observe that Mincut obtains the best overall performance. As we mentioned previously, Mincut requires the number of expected segments to be known *a priori*, which provides an advantage in document adaptation over algorithms that do not require such parameter. This advantage can be observed in Mincut's results consistency across domains while the other algorithms results oscillate more; given these circumstances, we will disregard Mincut's performance in the remaining of the result analyses. Looking at the results in each domain we can see that the best performing algorithm varies: Multiseg performs better in the Biography domain (a 0.05 WD difference the second best algorithm, Bayesseg-MD); CVS in the News domain (a 0.05 difference over MultiSeg); Bayesseg-MD in the AVL domain (a 0.01 difference to PLDA-MD); Bayesseg in the Physics domain (a 0.01 WD difference over CVS, PLDA-MD, and Bayesseg-MD).

	Biography	**News**	**AVL**	**Physics**
Random	0.52± 0.1	0.52±0.1	0.51±0.01	0.52±0.02
Mincut	0.39± 0.1	0.37±0.2	0.37±0.10	0.36±0.1
C99	0.61±0.2	0.49±0.3	0.59±0.20	0.54±0.2
CVS	0.54± 0.2	**0.42±0.2**	0.45±0.10	0.43±0.2
TextTiling	0.73± 0.2	0.63±0.2	0.47±0.10	0.49±0.2
Bayesseg	0.53±0.2	0.51±0.3	0.39±0.10	**0.42±0.2**
Bayesseg-MD	0.42±0.2	0.59±0.2	**0.37±0.10**	0.43±0.2
PLDA	0.58±0.2	0.53±0.3	0.55±0.20	0.49±0.2
PLDA-MD	0.54±0.2	0.54±0.3	0.38±0.10	0.44±0.2
MultiSeg	**0.37±0.2**	0.47±0.3	0.41±0.03	0.44±0.1

Table 5.2: Average WD scores. In bold, are the best results for each domain (excluding Mincut's results).

Looking at the results from the perspective of single- *vs.* multi-document models, we can see that the best approach is domain dependent. Single-document models perform better in the News domain (CVS with a 0.05 WD difference to the best multi-document model, MultiSeg) and in the Physics domain (Bayesseg with a 0.01 WD difference to Bayesseg-MD). Multi-document models perform better in the Biography, and AVL domains. The WD results improvements are 0.16 (comparing MultiSeg and Bayesseg), 0.23 (Bayesseg-MD and CVS), and 0.02 (Bayesseg-MD and Bayesseg), respectively. Comparing the Bayssseg-MD and PLDA-MD extensions with its single-document counterparts we can see improvements across most of the domains. The exception is the News domains (both Baysseg-MD and PLDA-MD) and the Physics domain (only Baysseg-MD). For the News domain, Baysseg-MD and PLDA-MD made the results worse by 0.08 and 0.01 WD margins. For the Physics domains, the difference between Baysseg-MD and Baysseg is 0.01. Despite multi-document models not strictly outperform single-document models, these results indicate that our hypothesis of using information from all documents to perform the segmentation task is a feasible strategy under specific scenarios.

Comparing the performance of the algorithms across domains, we can observe that generally worse WD scores are obtained in Jeong and Titov (2010) datasets. For example, TextTiling, Bayesseg, PLDA-MD have all of their worst results in these datasets. CVS, C99, and PLDA have a similar pattern with the difference that they are still able to perform well in the News domain. Another indicator of the underperformance of the algorithms in Jeong and Titov (2010) datasets is the number of results that are worse than the random baseline (16 out of 24). The exception to all the previous algorithms is MultiSeg, where the opposite occurred, that is, generally worse results were obtained in the AVL and Physics domains (the exception is the News domain).

5.3.3 Domain Analysis

To understand these domain differences we provide some document examples of the Biography, and News domains (Figures 5.3, and 5.4, respectively). From these documents, we can observe that most segments are short, and, consequently, there is no much opportunity for lexical cohesion to be observed. In the Biography document example, segments resemble a list of biographical facts instead of developing a topic in a more cohesive manner. For example, the document puts the marriage and separation of Princess Diana and Prince Charles in two distinct segments. These phenomenons continue to occur in the News domain where we can see short segments with a subtle topic change. For example, the second and third utterances of the News document mention how income is distributed among the different entities involved in the process of selling eBooks through the Google Editions platform but are in different segments. Again, these characteristics contribute to the results difference between them and the AVL and Physics datasets across the different segmentation algorithms.

```
==================================================================
Diana, Princess of Wales was one of the most famous women in the world.
Diana was born on 1 July 1961 as Diana Frances Spencer. Her father was Lord Spencer. She left school when she was 16 and moved to London when she was 17.
==================================================================
In 1981 Diana married Prince Charles at St. Paul's Cathedral. They had two sons, Prince William and Prince Henry.
==================================================================
Charles and Diana separated in 1992 and they divorced in 1996. Diana said Camilla Parker-Bowles was responsible for the problems with her marriage.
==================================================================
Princess Diana was well known for her charity work. She campaigned to end land mines. She also helped to make the lives of people with AIDS better.
==================================================================
Diana and her boyfriend, Dodi Al-Fayed, died in a car crash in Paris on 31 August 1997. Many people left flowers, candles, cards and personal messages for her in public places. She had a big funeral in London.
==================================================================
Diana's full title, while married, was Her Royal Highness The Princess Charles Philip Arthur George, Princess of Wales and Countess of Chester, Duchess of Cornwall, Duchess of Rothesay, Countess of Carrick, Baroness of Renfrew, Lady of the Isles, Princess of Scotland.
==================================================================
```

Figure 5.3: Example of a document in the Biography domain.

==

Google confirmed that Google Editions is ready for launch this summer. This is a 'buy anywhere, read anywhere' eBooks service that allows users to download eBooks on mobile phones, eBook readers and PC. Announced last year at the Frankfurt Book Fair, Google Editions will have about half a million eBooks available for purchase and download by late June or July.

Chris Palma, Google's manager of strategic-partner development, announced the time-table for Google's plans at the publishing-industry panel in New York yesterday, reported The Wall Street Journal. Google Editions will be an Amazon-like eBook store that will offer about 5,00,000 eBooks to users. Publishers get to keep 63 percent of income from the eBooks sold while Google retains 37 percent.

==

Users can also buy eBooks from Amazon as well as Barnes & Nobles through Google Editions. In that case, the publisher gets just 45 percent while Google gets to retain 55 percent of income. Apart from that, even independent book retailers would be allowed to sell Google Editions at their own sites.

Just when Apple is anticipating high growth of its iBook Store, Google is getting ready to roll out Google Editions. However, Google's idea is to access Google Editions from any browser and thus create an "open ecosystem" in the eBook market. Publishers will have a greater control over how their books are being sold.

==

Google has chosen the right time to launch Google Editions service with variety of tablets and mobile Internet devices emerging in the market. However, with Android on its side, we're sure Google will ensure something is packed in for Android OS phone based users.

==

For more on Google Editions, we'll have to wait till the end of June or July.

==

Figure 5.4: Example of a document in the News domain.

5.3.4 Document Modality Results

We also analyze the performance consistency of the algorithms according to the different document modalities in the AVL and Physics domains (Tables 5.3 and 5.4). For some of the algorithms we can observe that the results are modality consistent in both domains. For example, C99 and PLDA-MD perform better in the HTML and MultiSeg in video transcript documents; C99 obtains higher average WD results, ranging from 0.01 to 0.33, when compared to other modalities; PLDA-MD performs better with WD differences between 0.01 and 0.17; MultiSeg's result differences range from 0.01 to 0.11. It is also possible to observe the inverse pattern, that is, algorithms consistently underperforming in a modality. This is case of TextTiling, which performed worst in the video modality in both domains (differences between 0.29 an 0.15) and also CVS for HTML documents (differences between 0.26 an 0.02). For the Bayesseg and Bayesseg-MD the results were generally more balanced. The most noticeable exception are the video documents for the Physics domains where both algorithms performed worse than in the AVL domain, but, for the remaining modalities, the results difference is not as prominent. Regarding the PDF modality, which only exists in the Physics domain, it poses more difficulties to the PLDA and PLDA-MD algorithms since it is where they perform worst (both have a 0.08 WD difference to the second worst modality). Given these different types of tendencies, it is possible that interactions between the modality of a document and segmentation algorithms exist. Therefore, it is plausible that incorporating information targeting specific modalities improves the segmentation results.

	HTML	**PPT**	**Video**
C99	0.48±0.10	0.50±0.20	0.81±0.20
CVS	0.59±0.10	**0.38±0.10**	0.44±0.06
TextTiling	0.42±0.04	0.39±0.03	0.59±0.20
Bayesseg	0.41±0.04	0.39±0.20	0.39±0.01
Bayesseg-MD	0.41±0.05	**0.38±0.10**	**0.34±0.10**
PLDA	0.39±0.04	0.58±0.20	0.61±0.30
PLDA-MD	**0.35±0.04**	0.39±0.10	0.38±0.04
MultiSeg	0.43±0.03	0.41±0.03	0.40±0.01

Table 5.3: Average WD scores for each document modality in the AVL domain.

To understand if there is a bias of the segmentation algorithms to over/undersegment documents in different domains we look at the average segment count difference between hypothesis and reference segmentations in Table 5.5. Negative values indicate that the hypothesis contains less segments than the reference, while positive values indicate the opposite. The number of hypothesized segments is generally not far off from the number of segments in the references, which means that the WD are mostly influenced by bound-

	HTML	PPT	PDF	Video
C99	0.49±0.20	0.51±0.10	0.57±0.20	0.64±0.27
CVS	0.47±0.10	0.45±0.10	0.43±0.10	**0.33±0.20**
TextTiling	0.43±0.20	0.44±0.10	0.44±0.10	0.68±0.20
Bayesseg	**0.41±0.20**	0.43±0.10	**0.42±0.10**	0.47±0.20
Bayesseg-MD	**0.41±0.20**	**0.39±0.10**	0.44±0.10	0.49±0.20
PLDA	0.42±0.30	0.54±0.10	0.62±0.20	0.52±0.30
PLDA-MD	**0.41±0.10**	0.44±0.10	0.52±0.20	0.43±0.20
MultiSeg	0.47±0.06	0.45±0.05	0.44±0.05	0.36±0.10

Table 5.4: Average WD scores for each document modality in the Physics domain.

ary misplacement. Another observation is that being closer to the number of segments in the references is not always correlated with a better quality segmentation. One example is CVS, which has a closer number of segments in the News domain but the WD scores are higher. Looking at these results also allows us to understand what transformations occur in the segmentations when the multi-document model extensions are applied. For the PLDA case, we can observe that improvements can occur when PLDA-MD outputs more and less segments than PLDA. In the Biography domain PLDA-MD has more segments than PLDA, whereas in the AVL and Physics domains the opposite occurs. Similar interactions can be found between Bayesseg and Bayesseg-MD as well. This means that the multi-document improvements are not limited to always adding/removing segments since they are dynamic and adapt to the input data.

	Biography	News	AVL	Physics
C99	-4.8±5.3	-1.6±1.7	13.6±24.2	3.20±12.2
CVS	-0.8±1.1	-0.2±0.6	1.0±0.9	0.42±1.0
TextTiling	16.9±20.2	1.5±2.1	18.2±20.3	11.1±12.1
Bayesseg	-2.9±4.2	1.5±1.9	-1.6±5.6	1.50±3.6
Bayesseg-MD	2.70±3.9	3.3±2.5	-3.9±3.6	-1.5±1.9
PLDA	0.30±6.8	-0.1±2.0	13.0±28.3	2.8±8.3
PLDA-MD	3.60±6.8	0.6±2.4	-4.1±5.1	1.1±3.9
MultiSeg	1.90±3.1	1.8±1.9	-4.4±4.7	-0.3±2.4

Table 5.5: Average segment count difference between hypothesis and reference.

The overall conclusion is that it is indeed possible to improve results using multi-document segmentation. Despite this positive result, it should be noticed that multi-document approaches were not able to consistently get better results. The characteristics of the domain, particularly how lexical manifests and what the target segmentation is, play an important role. Therefore, some algorithms are a better fit for segmenting datasets with particular characteristics. A similar observation can be made regarding the results based on document modality since the result consistency of the models also varies along this dimension.

5.4 BeamSeg Segmentation

We now carry out segmentation experiments using BeamSeg and compare them with the previously established baseline results in order to determine if it is an effective approach to segmentation.

5.4.1 Experimental Setup

In the scope of the BeamSeg model, we want to investigate the role of two factors in segmentation: the impact of using the proposed dynamic language model prior *vs.* an independent language model prior, and the influence of the segment duration prior and its application based on modality. We perform experiments that test two different types of segment length prior: Beta-Bernoulli, and Gamma-Poisson. Each prior is tested by having a single prior variable for all documents in the dataset or by having individual variables conditioned on document modality. The experimental setup regarding datasets and hyperparameters setup follows the one described in Section 5.3.1.

5.4.2 Experimental Results

The following sections report and analyze BeamSeg's results in the available multi-document segmentation datasets (Jeong and Titov, 2010) and our learning materials dataset.

5.4.2.1 Available Datasets Results

We start by analyzing BeamSeg's results on the three datasets from Jeong and Titov (2010) (Table 5.6). For convenience, during the discussion of the results, we refer the language model prior as the LM prior, and the segment length prior as the SL prior. From the results, we can observe that no single combination of LM prior and SL prior obtained the best results in all three datasets. For the Biography domain, the best results were obtained when using a dynamic and Gamma-Poisson priors, which improved the WD average in 0.08 when compared to the independent LM prior version. The improvements stem from the dynamic LM prior being able to output more segments. The problem with the independent LM prior is that it outputs a single segment for 80% of the documents. This undersegmentation problem can also be observed in Table 5.7, where the average segment count difference between the hypothesis and reference are negative. By using a dynamic LM prior, the average number of segments increases 4.2, and the number of single-segment documents drops to 16.4%. Given the result improvements, we can conclude that some of these new segments are close to the reference. For the Beta-Bernoulli case, we can also see result improvements from the independent to the dynamic LM prior, although they are smaller, 0.01 in WD average. In this case, the differences in the number of segments and the number of single-segment documents are not as prominent as in the Gamma-Poisson prior. The segment count difference increases 1.3, and the number of single-segment document decreases 0.08%.

LM Prior	**SL Prior**	**Biography**	**News**
Independent	Beta-Bernoulli	0.54±0.2	0.46±0.3
	Gamma-Poisson	0.58±0.2	0.46±0.3
Dynamic	Beta-Bernoulli	0.53±0.2	0.47±0.3
	Gamma-Poisson	0.49±0.2	0.51±0.3
	Baseline	**0.37**±0.2	**0.42**±0.2

Table 5.6: BeamSeg average WD scores in Jeong and Titov (2010) datasets. The algorithms in the baseline results are MultiSeg, CVS, and Bayesseg-MD for the Biography, and News domains, respectively.

LM Prior	**SL Prior**	**Biography**	**News**
Independent	Beta-Bernoulli	-5.9±3.9	0.4±1.5
	Gamma-Poisson	-7.0±3.8	-1.9±1.2
Dynamic	Beta-Bernoulli	-4.6±3.4	-1.1±1.4
	Gamma-Poisson	-2.8±3.6	-1.4±1.9

Table 5.7: BeamSeg average segment count difference between hypothesis and reference.

In the News dataset, we see that moving from an independent to dynamic LM prior can make results worse. This occurs in the Gamma-Poisson and Beta-Bernoulli priors with 0.05 and 0.01 WD increases, respectively. This is related to the number of single-segment documents. In the Beta-Bernoulli prior, a 56.9% increase of single-segment documents can be observed. For the Gamma-Poisson, the number of single-segment documents is already high (91.8%) for the independent LM prior, and the trend carries over to the dynamic version (84.1%). This shows that the dynamic prior is not able to output more segments in the News domain, and, thus, improvements similar to the Biography domain could not be obtained.

From the previous results, we can see that BeamSeg does not perform better than the baseline results obtained in Section 5.3. The differences in WD to the best baseline results are 0.12, and 0.05 for the Biography, and Newsdomains, respectively. Therefore, it is not a suitable approach for these domains where there is not much topic development within the document segments.

5.4.2.2 Learning Materials Datasets Results

We now carry out a result analysis of the AVL and Physics datasets. Recalling that these datasets have documents from four possible modalities (HTML, PPT, PDF, and video), we study how applying the SL prior conditioned on document modality influences segmentation. Table 5.8 shows the obtained average WD results. A first observation is the fact that different SL prior scopes work better with particular types of LM prior. For example, in the Beta-Bernoulli and Gamma-Poisson cases, the dynamic LM prior always has better results when using a SL prior conditioned to the document modality than with using the same prior for the whole dataset. The WD improvements are 0.06 and 0.12 for the AVL and Physics domains, respectively. Looking at the results of the independent LM prior we see that the WD scores of the modality SL prior are generally worse than the ones obtained when using the dataset version. For the Gamma-Poisson, the results are 0.03 and 0.01 higher, in the AVL and Physics domains, respectively. Similar behavior can be seen with the Beta-Bernoulli in the AVL domain were the WD increases 0.01 (in the Physics domain, the results actually improve by a 0.03 WD margin). The main difference between reference and hypothesis segmentations is that the reference contains more segments (Table 5.9). This is a general tendency for all the tested prior configurations. From this data, we can also observe that some prior configuration exhibit a particular pattern. For example, when using an independent LM prior with the Gamma-Poisson, the modality scope increases the number of obtained segments. Switching to the dynamic LM prior we can see the opposite effect, the number of segments decreases. Therefore, one of the reasons for this particular prior configuration to obtain the best WD results is its ability to remove incorrect segments. In the Beta-Bernoulli case, the behavior is different, as the segment count difference always decreases. This shows that different types of interactions between the priors exist and these influence the obtained segmentation in different ways, and, thus, choosing a suitable configuration is essential to obtain the best results.

Given the close scores of Baysseg-MD and BeamSeg in the AVL domain, we analyze the individual WD document scores in Table 5.10. From these results, we can see that BeamSeg has better results in five out of the ten documents, one tie, and two documents where the WD difference is only 0.01. This leaves Bayesseg-MD to perform significantly better than BeamSeg only in two test cases. Therefore, despite the close WD average scores, BeamSeg is more consistent than Bayesseg-MD in the majority of the test cases.

Similarly to the AVL domain, we also carry out a more fine-grained analysis of the WD results of BeamSeg and the Bayesseg baseline in the Physics domain. Taking into account that the number of documents in the Physics domains is high, we aggregate the results by subject. In Table 5.11, we observe that BeamSeg performs better in three subjects, has worse results in two subjects, and the same results in the other two subjects. At the document level, BeamSeg has better results in 66 documents, worse results than Bayesseg

LM Prior	SL Prior	Scope	AVL	Physics
Independent	Beta-Bernoulli	D	0.39±0.10	0.45±0.10
		M	0.40±0.10	0.42±0.10
	Gamma-Poisson	D	0.40±0.10	0.41±0.10
		M	0.43±0.10	0.42±0.10
Dynamic	Beta-Bernoulli	D	0.44±0.10	0.54±0.20
		M	0.38±0.10	0.42±0.10
	Gamma-Poisson	D	0.38±0.10	0.47±0.20
		M	**0.37±0.10**	**0.40±0.10**
		Baseline	**0.37±0.10**	0.42±0.20

Table 5.8: BeamSeg dataset prior average WD scores. The scope column indicates if the SL prior was applied based on the documents' modality (M) or if it was the same for the whole dataset (D). The algorithms in the baseline results are Bayesseg-MD, and Bayesseg for the AVL and Physics domains, respectively.

LM Prior	SL Prior	Scope	AVL	Physics
Independent	Beta-Bernoulli	D	-4.3±2.7	-1.4±2.5
		M	-5.7±2.6	-2.3±2.3
	Gamma-Poisson	D	-5.9±2.9	-2.7±2.1
		M	-4.9±2.4	-0.3±6.1
Dynamic	Beta-Bernoulli	D	-0.8±4.3	3.3±6.7
		M	-4.7±3.3	-1.1±2.9
	Gamma-Poisson	D	-1.8±4.3	0.8±4.4
		M	-4.9±4.2	-2.3±1.9

Table 5.9: BeamSeg average segment count difference between hypothesis and reference.

in 51 documents, and the same results as Bayesseg in the other 34 documents. Therefore, BeamSeg has 15 more documents where it outperforms Bayesseg. This indicates that the result improvements are spread across different test cases rather than having a large result difference in a particular test case.

To have a better grasp how the scope of SL prior (dataset or modality) influences the results, we present the average WD scores aggregated by document modality in Tables 5.12 and 5.13. When comparing the WD results of the dynamic LM prior we can see that there is an overall tendency for the WD to improve when using the modality-based prior instead of the dataset prior. The only exception are PPT documents in the AVL domain when using the Gamma-Poisson prior. In this case, the best results are actually obtained by the independent LM prior with a 0.04 WD difference. This does not happen in the Physics domain where the pattern of using the dynamic LM with a modality-based Gamma-Poisson obtains the best results for PPT documents. Also in the Physics domain, we can observe that for the video modality the best results occur when using an independent LM prior and a SL prior at the dataset level (a 0.08 WD difference to

		WD	
	Modality	**Bayesseg-MD**	**BeamSeg**
Doc_1	HTML	0.49	**0.42**
Doc_2	HTML	0.39	**0.30**
Doc_3	PPT	**0.29**	0.37
Doc_4	PPT	**0.39**	0.40
Doc_5	PPT	0.48	**0.36**
Doc_6	PPT	0.43	**0.29**
Doc_7	PPT	**0.40**	0.56
Doc_8	Video	**0.27**	0.28
Doc_9	Video	0.42	**0.34**
Doc_{10}	Video	**0.42**	**0.42**

Table 5.10: WD document scores in the AVL domain.

Subject	**Bayesseg**	**BeamSeg**
$L02$	**0.39±0.16**	**0.39±0.09**
$L03$	0.45±0.16	**0.42±0.09**
$L06$	0.45±0.15	**0.41±0.10**
$L08$	**0.36±0.16**	**0.36±0.11**
$L10$	**0.38±0.15**	0.41±0.06
$L11$	**0.35±0.25**	0.43±0.22
$L20$	0.48±0.17	**0.43±0.14**

Table 5.11: WD scores aggregated by subject in the Physics domain.

the dynamic LM prior version). Based on the previous result improvements when using a modality-based SL prior, we corroborate our hypothesis that documents sharing the same modality have similar segment length characteristics that are worth abstracting on the segmentation model. We also conclude that in order to obtain these modeling advantages it is necessary to use suitable priors, which we determined to be the dynamic LM prior and a Gamma-Poisson SL prior.

Summing up the conclusions of the previous experiments, one of the key observations is that the performance of the segmentation models depends on the lexical cohesion characteristics of the domain. This is corroborated by the fact that three different segmentation algorithms performed better in the four tested domains. Two of the algorithms are multi-document models, which makes a favorable argument for using this approach over single-document segmentation. BeamSeg turned out to not be suitable for domains where there is not much topic development in each segment (the Biography, and News domains), and, consequently, lexical cohesion is not prominent. The AVL and Physics domains have different characteristics; the segments are longer, making more room for lexical cohesion to be formed, and relate to each other. In this setup, BeamSeg's modeling assumptions are effective in obtaining state-of-the-art results. In the AVL

LM Prior	SL Prior	Scope	HTML	PPT	Video
Independent	Beta-Bernoulli	D	0.43±0.01	**0.37±0.10**	0.40±0.03
		M	0.43±0.01	0.39±0.10	0.39±0.01
	Gamma-Poisson	D	0.41±0.02	0.41±0.1	0.39±0.01
		M	0.42±0.05	0.39±0.08	0.51±0.10
Dynamic	Beta-Bernoulli	D	0.39±0.10	0.45±0.10	0.42±0.05
		M	0.38±0.04	0.39±0.1	0.39±0.01
	Gamma-Poisson	D	0.37±0.10	0.39±0.10	0.36±0.02
		M	0.36±0.10	0.41±0.10	**0.31±0.03**
		Baseline	**0.35±0.04**	0.38±0.10	0.34±0.10

Table 5.12: BeamSeg average WD scores for each document modality in the AVL trees domain. The algorithms in the baseline results are PLDA-MD (HTML), and Bayesseg-MD (PPT and Video). It should be noted that for PPT documents CVS obtained a similar WD average.

LM Prior	SL Prior	Scope	HTML	PPT	PDF	Video
Independent	Beta-Bernoulli	D	0.44±0.10	0.46±0.10	0.48±0.10	0.45±0.20
		M	0.41±0.10	0.43±0.10	0.47±0.10	0.39±0.20
	Gamma-Poisson	D	**0.40±0.10**	0.42±0.10	0.47±0.10	**0.35±0.20**
		M	**0.40±0.10**	0.41±0.10	0.57±0.10	0.38±0.10
Dynamic	Beta-Bernoulli	D	0.51±0.20	0.53±0.10	0.50±0.10	0.64±0.20
		M	0.41±0.10	0.42±0.10	0.46±0.10	0.41±0.20
	Gamma-Poisson	D	0.43±0.10	0.48±0.10	0.55±0.30	0.47±0.10
		M	**0.40±0.10**	**0.39±0.10**	**0.40±0.10**	0.43±0.20
		Baseline	0.41±0.10	**0.39±0.10**	0.42±0.10	**0.33±0.20**

Table 5.13: BeamSeg average WD scores for each document modality in the Physics domain. The algorithms in the baseline results are Bayesseg-MD (HTML and PPT), Bayesseg (PDF), and CVS (Video). It should be noted that for the HTML documents Bayesseg and PLDA-MD obtained a similar WD average.

domain, the average WD results are similar to the best baseline (Bayesseg-MD) and is able to obtain better results in more individual test cases. In the Physics domain, the average WD results improve 0.02 compared to the best baseline (Bayesseg) and obtains best results in 15 more documents than the baseline. Choosing an effective SL prior and applying it on a document's modality-basis is crucial to obtain the best results. According to our experiments, the Gamma-Poisson is a better prior than the Beta-Bernoulli. Under these conditions, we can observe the positive impact that the dynamic LM prior modeling assumption has in the results. The improvements in average WD are 0.03 and 0.02 when compared with the independent LM prior version, for the AVL and Physics domains, respectively. This indicates that the BeamSeg model is able to model datasets where document segments are related to each other by assuming that the underlying language models have a dynamic structure instead of being independently drawn from a Dirichlet distribution.

5.4.3 Qualitative Analysis

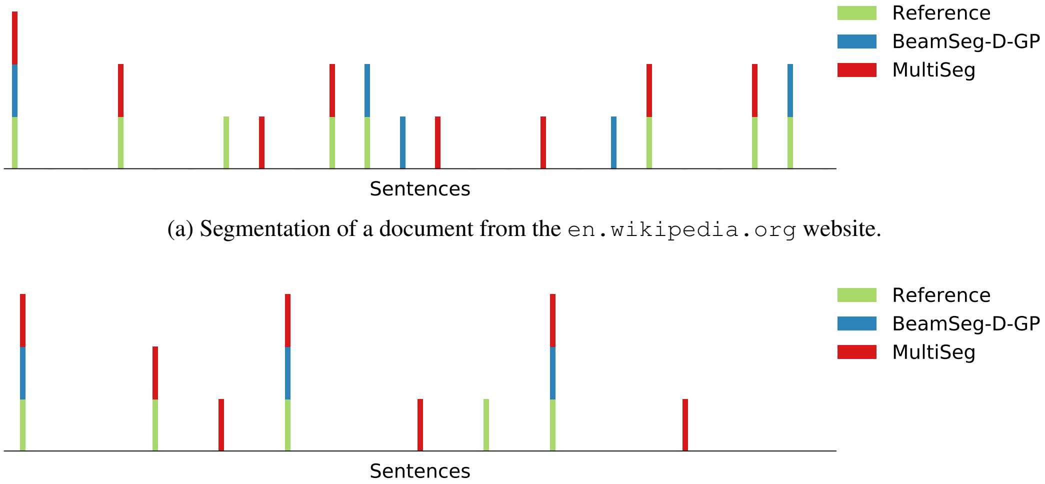
In this section, we perform a qualitative analysis of the segmentations obtained using the BeamSeg model. The goal is to understand in more detail how BeamSeg behaves in different documents with the different LM and SL prior configurations. To this end, we visually represent the document segmentations and describe the underlying content of the segments.

5.4.3.1 Biography Domain

Figure 5.5 shows two segmentation examples in the Biography domain obtained using BeamSeg and MultiSeg (the best performing algorithm in this domain). These examples illustrate the general segmentation pattern of the algorithms and the corresponding segmentation quality differences. BeamSeg outputs fewer segments than MultiSeg, which is a disadvantage in this domain since it is characterized by documents with a high number of short segments (Figure 5.5a). The test cases where BeamSeg is able to outperform MultiSeg are documents with a lower number of segments and when MultiSeg outputs many non-existent boundaries (Figure 5.5b). It should be noticed that, despite the provided examples showing that both BeamSeg and MultiSeg are generally precise in their boundary placement, it is also possible to find degenerated segmentations where boundaries are incorrectly placed by both algorithms. When comparing the content of the segments we can see that sometimes a topic has a dedicated segment in a document and others it is part of a larger segment. Examples of this situation can be found in the Barack Obama biography documents from Figure 5.5. For instance, the fifth segment in Figure 5.5a describes a book published by Barack Obama. This topic is also referred in the fourth segment of the document in Figure 5.5b, where the main topic is Barack Obama's transfer to Columbia University, which is also the topic of the fourth segment of the document in Figure 5.5a. MultiSeg deals with this aspect of the target segmentation by assigning topics that only occur in that particular topic (local topics). In theory, BeamSeg can also output local topics, but not explicitly modeling them makes the model prefer having a larger segment in this situation.

5.4.3.2 News Domain

As mentioned before, in the News domain, BeamSeg outputs single-segment documents in the majority of the test cases. Moreover, inspecting segmentations within the same subject shows that BeamSeg also assigns the same topic label to the majority of the segments. In Figure 5.6a, we can see a document where BeamSeg failed to identify most of the segment boundaries. This is a document describing how people are not careful in protecting their private information on social media. The segment that BeamSeg identifies talks

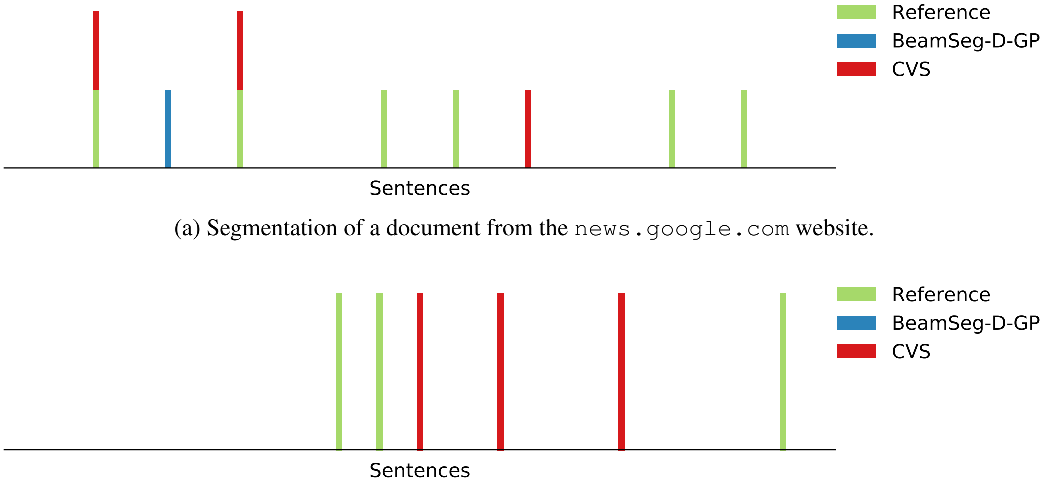


(a) Segmentation of a document from the `en.wikipedia.org` website.

(b) Segmentation of a document from the `biography.com` website.

Figure 5.5: Segmentation examples in two different documents about Barack Obama.

about a survey related to this private information topic and the following one is advise for people to protect themselves regarding this issue. In the reference, the advice is split into a list of bullet points originating segments where lexical cohesion does not develop much. Given the previous condition, BeamSeg identifies most of the documents in a subject with the same topic. The single-document model CVS, which obtained the best results in this domain, is able to identify these nuanced segment boundaries better. It should be noted though that it is also possible to find degenerated segmentations output by CVS corresponding to single-segment documents or with frequent boundary misplacement, such as the one in Figure 5.6b.



(a) Segmentation of a document from the `news.google.com` website.

(b) Segmentation of a document from the `news.google.com` website.

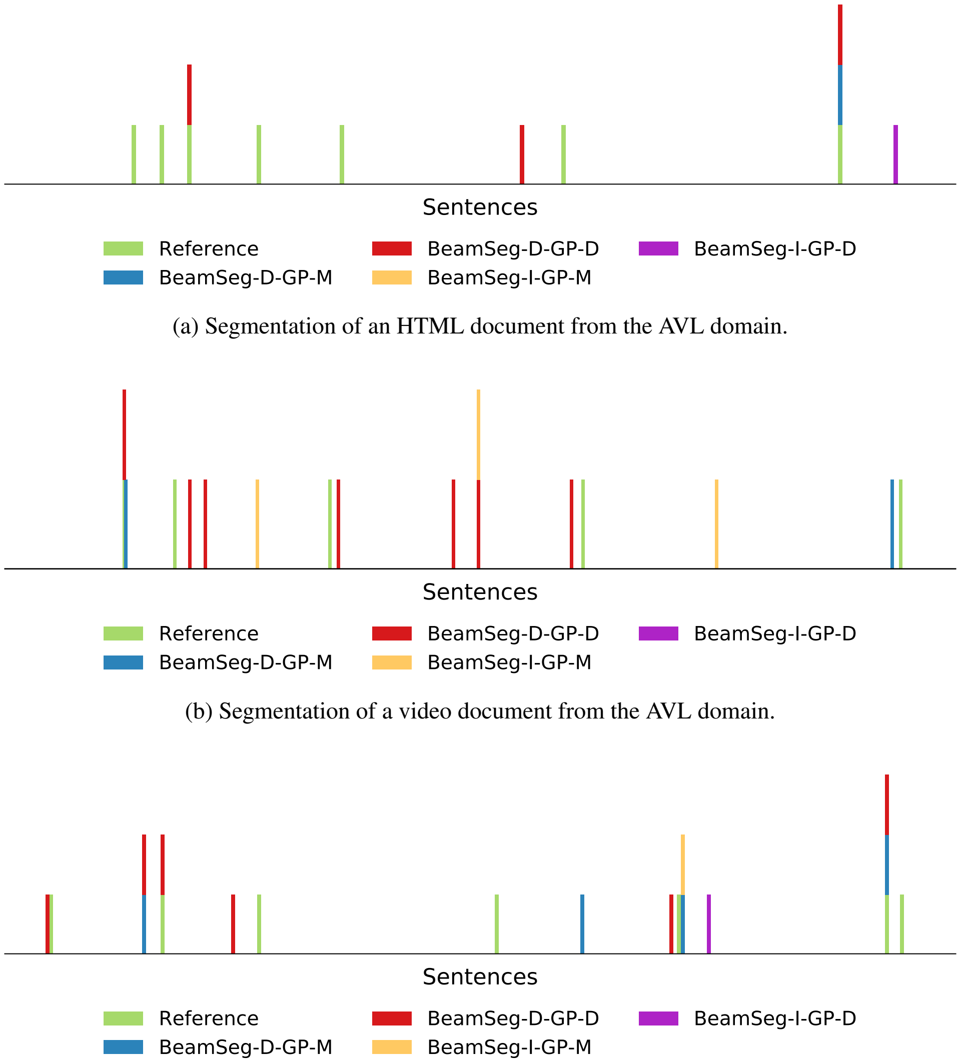
Figure 5.6: Segmentation examples in two different documents from the News domain.

5.4.3.3 AVL Trees Domain

We now focus on the segmentation differences of the various prior configurations tested with BeamSeg across different document modalities in the AVL domain. Segmentation examples can be found in Figure 5.7, where the prior configurations were abbreviated as follows: the first letter defines the type of LM prior – independent (I) or dynamic (D) – GP refers to the Gamma-Poisson SL prior, and the last letter to the scope of the SL prior – dataset (D) or modality (M). Looking at the segmentations, we can see that the major difference between the independent and dynamic LM priors is the number of output segments. When using the independent LM prior the number of obtained segments is low, especially when using the SL prior at the dataset level. This can be seen in Figures 5.7a and 5.7c. For the modality-based SL prior, longer documents have more segments, but these tend not to be accurate boundaries (Figure 5.7b). When using the dynamic LM prior the number of segments increases throughout all test cases. We can also observe different behaviors at the SL prior level. The dataset-based SL prior makes BeamSeg output more segments than the modality-based version. In some cases, like the segmentation in Figure 5.7a, it does allow to discover accurate segments that the modality prior does not detect. However, in the majority of the cases it ends up placing many non-existent boundaries such as in the segmentation of Figure 5.7b. Despite the modality-based SL prior not finding that many segments, the identified ones tend to be accurate. This makes sense since the over-segmentation of the dataset SL prior might be related with the bias it creates towards documents with short segments and the modality prior is able to adjust to a wider variety of documents, and, thus, originate segments based on language usage. Looking at the topics in the segments identified by BeamSeg, using the best prior configuration (BeamSeg-D-GP-M), we observe several situations where the fix topic ordering assumption impacts negatively the segmentation discovered by model. We observe that for some document a topic segment is correctly identified, while in others it is put together with other topics in a large segment. This is related to the fact that some documents share at least some of the topic order, while others have a very different one. For such scenarios, BeamSeg merges segments if the language does not change significantly. An example of this situation is the AVL trees node deletion topic (second to last segment in Figure 5.7b and third to last segment in Figure 5.7c in the reference segmentation). For the former case, BeamSeg was not able to identify the topic segment whereas in the latter it identified the topic correctly.

5.4.3.4 Physics Domain

The segmentations in the Physics domain maintain the same patterns of the AVL domain, reinforcing the previous findings at a larger scale. Figure 5.8 show different examples for each of the tested modalities. In the examples, we observe that independent LM priors output fewer segments than dynamic LM priors. Also,



(a) Segmentation of an HTML document from the AVL domain.

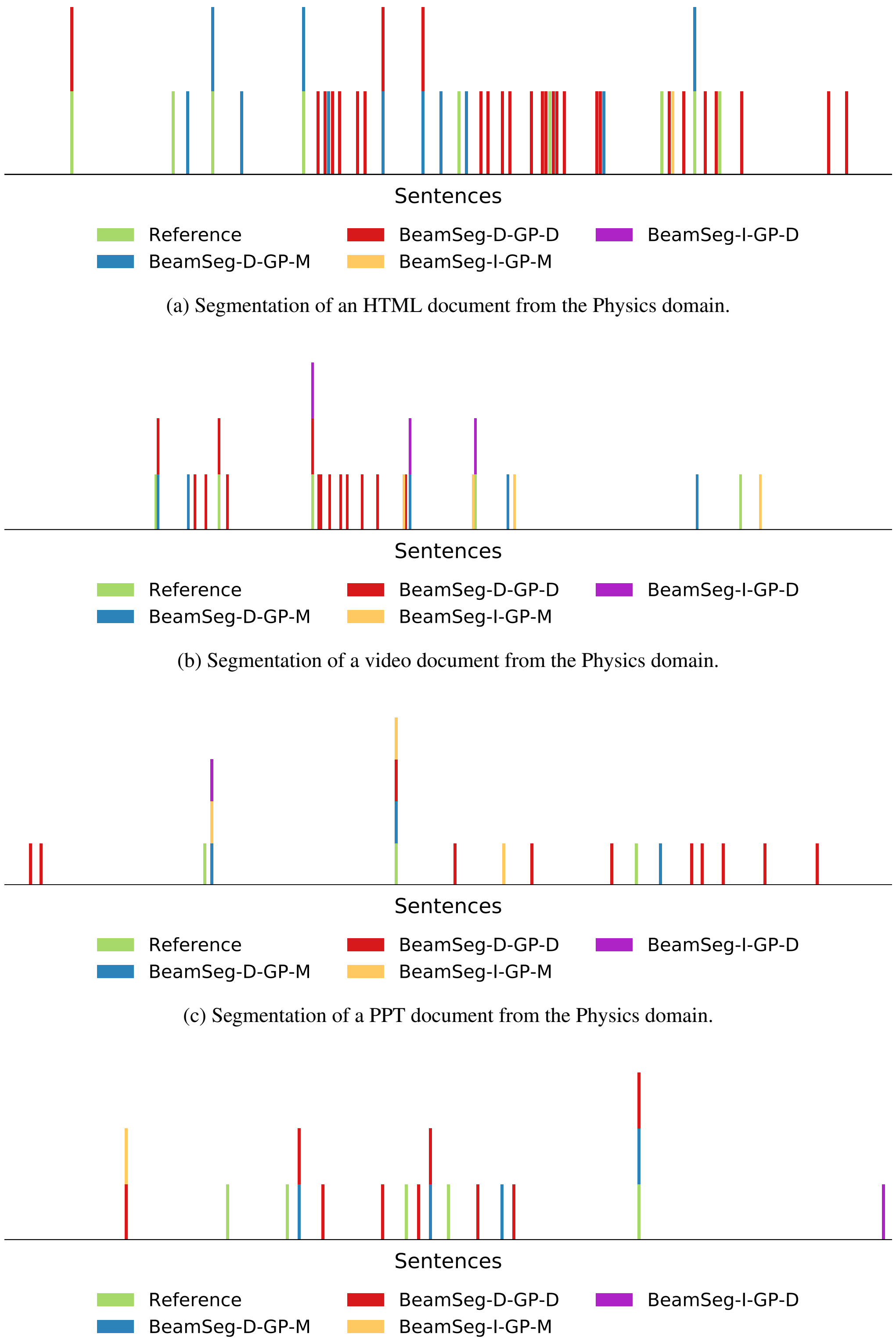
(b) Segmentation of a video document from the AVL domain.

(c) Segmentation of a PPT document from the AVL domain.

Figure 5.7: Document segmentation examples.

the dataset SL prior oversegments documents while the modality-based is more precise in the segmentation. One of the differences in the Physics domains is that we find more equations. The variables of the equations function as regular words, and since they get repeated constantly during a formula derivation that can span through several sentences, BeamSeg mistakes them for a segment. This is the case for some of the extra segments placed by BeamSeg in the fifth segment of the reference in Figure 5.8a. The problem of the topic order can also be observed in this domain. For example, BeamSeg identifies a segment describing instantaneous velocity in Figure 5.8 (the second segment) but there are other documents where this topic is merged in a larger segment. This problem of topic order is exacerbated in the Physics domain since the topic intersections between documents can be very different.

With the presented qualitative analysis of BeamSeg segmentations in all of the tested domains, we were able to observe how different factors impact the results. The interplay between domain characteristics, the target segmentation, and the modeling assumptions at the prior level in the BeamSeg model impacts the output segmentations. Therefore, choosing the correct prior assumptions according to what is expected to occur in the data is important to be able to obtain a segmentation that can be as accurate as possible.



(a) Segmentation of an HTML document from the Physics domain.

(b) Segmentation of a video document from the Physics domain.

(c) Segmentation of a PPT document from the Physics domain.

(d) Segmentation of a PDF document from the Physics domain.

Figure 5.8: Document segmentation examples.

6 Topic Identification Evaluation

In this chapter, we want to determine if a joint model of segmentation and topic identification improves the results of a pipeline strategy (performing the two tasks sequentially). Therefore, we compare graph-community detection algorithm results (Section 6.1) with the ones obtained by BeamSeg (Section 6.2).

6.1 Pipeline Approach to Topic Identification

In Section 4.3, we proposed a graph-community detection algorithm for topic identification that finds word communities and assign segments with a scoring function. In this section, we evaluate this approach.

6.1.1 Experimental Setup

We survey the following graph-community detection algorithms: Label Propagation (LP) (Raghavan et al., 2007), **Bigclam** (Yang and Leskovec, 2013), **CNM** (Clauset et al., 2004a), **Louvain** (Blondel et al., 2008), **Leading Eigenvector** (Newman, 2006), **Fast Greedy** (Clauset et al., 2004b), and **Walktraps** (Pons, 2006). Since topic identification can be framed as a problem of clustering segments, we use as a baseline for the pipeline strategy the following clustering algorithms:

Agglomerative clustering (Maimon and Rokach, 2005): a hierarchical clustering that uses a bottom-up approach, considering, in the beginning, all points as individual clusters. The procedure consists of a series of iterations, and, in each of them, two clusters are merged. Therefore, in each step, it is necessary to decide which clusters to merge, which is done by using a similarity measure and criterion function.

k-means (Lloyd, 2006): given a k value that specifies the number of target clusters, the process is based on minimizing a criterion function measures the distance of data points and cluster centroids.

DBSCAN (Sander et al., 1998): based on 2 values, a radius (Eps) and minimum number of neighbors ($MinPts$). When a point in a Eps-neighborhood contains at least $MinPts$ it is defined as a *core* point. When a core point is found, a cluster is formed with all its neighbors. Then, an expansion process takes place by checking if the neighbors are also core points. In the positive case, these are added to the cluster.

Spectral Clustering (Weiss, 1999): uses an adjacency matrix of a similarity graph between the points to cluster. Then, the first k eigenvectors of the Laplacian matrix are calculated. The final step consists of using the k-means algorithm to obtain a clustering from the eigenvectors.

We use the development sets from the segmentation experiments to tune the parameters of all algorithms. For the clustering algorithms that require the target number of clusters (Agglomerative, k-means, and Spectral), we use the number of topics in the reference. In our graph-community approach, we proposed several weighting schemes to study its impact on the discovered communities. We also proposed several scoring functions that assign segments to the discovered communities. We use the weighting schemes and scoring functions that yield the best results in the development set. Following Shahaf et al. (2012), we also use a *tf-idf* filter to prune words that are either too common or too rare to be representative of a topic. We experiment in the development set with a range between 1 and 200 for the cutoff value.

The pipeline strategy assumes that a segmented dataset is given as input to the topic identification algorithm. To have the best possible baseline, we use the golden standard segmentation as input for the algorithm in the pipeline strategy. The experiments are carried out in the same datasets used in the segmentation experiments. We use the B^3 score as an evaluation metric given that this experiment has a similar setup to the topic identification agreement study (Section 3.4).

6.1.2 Experimental Results

From the average B^3 results in Table 6.1 we can observe that the clustering algorithms that require the target number of clusters (Agglomerative, k-means, and Spectral) obtain the best results in all domains. Comparing these results with the best performing approaches that do not require this parameter, we obtain the following differences in B^3: 0.08 (Biography), and 0.03 (News), and 0.13 (AVL and Physics). This shows that knowing the number of target clusters brings performance advantages in the topic identification task. Connecting this with the observations made in the human agreement study (Section 3.4), where annotators in some situations grouped or split the same set of segments in different topics, it is plausible that a topic granularity also exists. This is similar to the segmentation granularity situation where Mincut, which requires the target number of segments (granularity), obtained the best performance. Comparing the results of the algorithms which do not require a target number of clusters, we can see that the best approach varies across domains. DBSCAN performs better in the Biography, and Louvain in the News, AVL, and Physics domains.

	Algorithm	**Biography**	**News**	**AVL**	**Physics**
GCD	Bigclam	**0.57±0.06**	0.60±0.12	0.31	0.36±0.05
	Leading Eigenvector	0.56±0.09	0.63±0.11	0.33	0.34±0.04
	Label Propagation	0.43±0.12	0.59±0.09	0.34	0.29±0.05
	Louvain	**0.57±0.08**	**0.64±0.11**	**0.37**	**0.38±0.06**
	Fast Greedy	0.55±0.08	**0.64±0.10**	0.33	0.35±0.07
	Walktraps	0.47±0.10	0.63±0.09	0.36	0.34±0.05
	CNM	0.54±0.05	**0.64±0.12**	0.33	0.34±0.04
Clustering	Agglomerative	**0.72±0.08**	0.66±0.09	0.39	0.46±0.03
	K-means	0.59±0.05	0.66±0.09	0.48	**0.49±0.02**
	Spectral	0.61±0.07	**0.67±0.12**	**0.50**	0.37±0.02
	DBSCAN	0.66±0.08	0.60±0.09	0.33	0.34±0.06

Table 6.1: Average B^3 scores for the topic identification task. In bold, are the best results for each class of algorithm, graph-community detection (GCD) or clustering.

The identified topics in the clustering and graph-community detection approaches have different characteristics. What mainly defines them is the number of identified topics and how segments are distributed. From Table 6.2 we see that it is possible that the same algorithm discovers both a low and higher number of topics compared to the reference in different domains. For example, DBSCAN identifies 269 more topics in the Biography domain and identifies 88 fewer topics in the News domain. When the number of topics is high, the segment distribution has a long tail of topics with a single segment (local topics). This is a source errors since the number of local topics does not match the reference. When the number of identified topics is low, we observe that there is a small number of topics that contains the majority of the segments, causing another type error. It should be noticed that there is no strict correlation between the number of identified topics and the best results. For example, in the Biography domain, DBSCAN is the algorithm that identifies the highest number of topics and obtains the best performance. This means that DBSCAN is able to correctly group segments and isolate segments for which the topic cannot be identified. For the Physics domain, Louvain obtains the best performance. In this domain, DBSCAN maintains its topic segment distribution behavior, entailing that the topics with a higher number of segments are incorrectly clustered.

Taking into account that the joint model approaches do not require a target number of topics, we want to compare them with algorithms with similar characteristics. Therefore, we consider the results of the following algorithms as a baseline: DBSCAN (Biography), and Louvain (News, AVL, and Physics). Given that three out of five of the baselines correspond to the results of a graph-community detection-based algorithm, it indicates that this approach was effective.

		Biography		News		AVL		Physics	
	Algorithm	T	L	T	L	T	L	T	L
Clustering	Agglomerative	405	119	220	110	17	13	135	99
	k-means	405	331	220	104	17	15	135	113
	Spectral	405	151	220	52	17	13	135	43
	DBSCAN	674	481	132	110	4	0	454	420
GCD	Bigclam	584	389	209	154	45	32	325	214
	Leading Eigenvector	189	10	209	37	5	0	32	0
	Label Propagation	131	18	121	46	3	0	16	6
	Louvain	200	14	216	47	7	0	48	4
	Fast Greedy	190	14	225	67	5	1	27	1
	Walktraps	274	99	220	70	6	0	98	28
	CNM	234	54	209	50	4	0	30	2
	Reference	405	112	220	50	17	5	135	53

Table 6.2: Number of total and local identified topics (T and L columns) by the clustering and graph-community detection algorithms. Local topics represent topics that are only assigned one segment.

6.2 Joint Model Approach to Topic Identification

We now describe the topic identification experiments using a join model approach. The goal of the experiment is two-fold: study how different BeamSeg prior combinations perform in this task and determine if better results than the pipeline approach can be obtained.

6.2.1 Experimental Setup

The experimental setup is similar to the pipeline approach. We use the same datasets and the B^3 metric for evaluation. We also evaluate the topic identification performance of MultiSeg, which is, to the best of our knowledge, the only joint model for segmentation and topic identification in the literature.

6.2.2 Experimental Results

We first analyze the results in Jeong and Titov (2010) datasets (Table 6.3). The B^3 scores show that none of the joint models obtains better results than the pipeline approach. The differences to the baseline are 0.12, and 0.02 for the Biography, and News domains, respectively.

Comparing the joint models, we can see that BeamSeg always performs better. In the Biography and News domain, BeamSeg outperforms Multiseg by 0.02 and 0.16. From the BeamSeg results, we can also observe that different prior configuration work better in different domains. For example, the dynamic LM priors perform better in the News domain but in the Biography domain the independent LM is better. Overall, using the Beta-Bernoulli affords better results. The exception is the News domain. Relating these results to the segmentation task, we conclude that performing worse in the segmentation task might not translate to worse topic identification results. This is the case of the Biography and News domains, where MultiSeg obtains better segmentation but BeamSeg affords better topic identification.

LM Prior	**SL Prior**	**Biography**	**News**
Independent	Beta-Bernoulli	0.51±0.11	0.60±0.10
	Gamma-Poisson	0.37±0.10	0.62±0.10
Dynamic	Beta-Bernoulli	0.54±0.06	0.57±0.09
	Gamma-Poisson	0.53±0.06	0.60±0.08
	MultiSeg	0.52±0.11	0.43±0.12
	Baseline	**0.66±0.08**	**0.64±0.11**

Table 6.3: Joint models B^3 scores in Jeong and Titov (2010) datasets.

In Table 6.4, we can see the differences between the joint model approaches in the number of identified topics and the reference. The same behavior patterns observed in the pipeline experiment also occur in this case. That is, test cases where the number of identified topics is much higher than the reference also have a high number of local topics. This occurs in MultiSeg, which for all domains identified more topics than the reference. Such a high number of local topics explains the worse performance of MultiSeg when compared to BeamSeg in the topic identification task since each local topic that is not in the reference does not have relations to other segments. This contrasts with BeamSeg which identifies fewer topics and clusters more segments in each topic, affording a better topic identification performance. Comparing the identified topics between different prior configurations, we can see different behaviors across the domains. For the Biography domain, using the dynamic prior allows more topics to be identified when compared to the independent prior, which better matches the reference. This correlates with the differences in segmentation described previously. It turns out that the dynamic LM prior is making BeamSeg output many single segment document, many of which originate a local topic. When using the independent version, more segments can be obtained and correctly assigned the same topic, providing better topic identification results.

Table 6.5 depicts the average B^3 scores of the joint models in the topic identification task. BeamSeg obtains the best results, improving the B^3 score of the pipeline approach by 0.02 and 0.03, for the AVL and Physics domains, respectively. The result improvements are higher when compared with MultiSeg, 0.1 for both domains. Comparing the results of different prior configurations, we can see that, in the AVL

		Biography		**News**	
LM Prior	**SL Prior**	T	L	T	L
Independent	Beta-Bernoulli	130	21	336	103
	Gamma-Poisson	76	22	137	64
Dynamic	Beta-Bernoulli	239	101	206	127
	Gamma-Poisson	289	79	203	130
	MultiSeg	734	264	542	70
	Reference	405	112	270	50

Table 6.4: Total number of identified topics by the joint model approaches.

domain, three different configurations obtain the best results. For the Physics domain, a higher range of result differences can be observed. From these differences, we can see that the Beta-Bernoulli prior works better at the modality scope and when using an independent LM prior. This is different in the Gamma-Poisson case, where the best performance is achieved when using the SL prior at the modality level with the dynamic LM prior. Therefore, particular interactions between the different types of priors exist, and, thus, choosing the correct configuration is critical to obtain the best performance in the topic identification task. This relates to the previous segmentation results where the performance patterns are similar.

LM Prior	**SL Prior**	**Scope**	**AVL**	**Physics**
Independent	Beta-Bernoulli	D	0.35	0.36±0.05
		M	**0.39**	0.38±0.06
	Gamma-Poisson	D	0.38	0.35±0.03
		M	0.36	0.37±0.05
Dynamic	Beta-Bernoulli	D	**0.39**	0.30±0.04
		M	0.32	0.34±0.05
	Gamma-Poisson	D	0.38	0.31±0.02
		M	**0.39**	**0.41±0.06**
		MultiSeg	0.29	0.30±0.03
		Baseline	0.37	0.38±0.06

Table 6.5: Joint models B^3 scores in the AVL and Physics domains.

From the number of topics identified by the joint models in Table 6.6, we can see that MultiSeg behaves differently in these domains. The difference is that now it identifies fewer topics than the ones in the references and most of them are not local topics. The distribution of the segments among the topics is balanced with many segments being assigned to the same topics. Given MultiSeg's underperformance, this means many segments incorrectly share the same topic. In what respects BeamSeg, the best prior configuration has a number of identified topics closer to the reference than MultiSeg but it still identifies fewer topics. This BeamSeg configuration identifies 7 and 65 fewer topics than in the references for the AVL and Physics domains, respectively. Other prior configurations are able to get a closer number of identified topics to reference but the B^3 scores are worse. For example, using a dataset scope instead of modality in the Physics domain allows for topics to be identified with a difference of just two topics to the reference. The problem is that the dataset prior originates more segments than the ones in the reference, which ends up being a source of topic identification errors. When using the modality SL prior the number of segments is lower and the topic identification is more accurate, explaining the better results.

			AVL		**Physics**	
LM Prior	**SL Prior**	**Scope**	T	L	T	L
Independent	Beta-Bernoulli	D	9	2	80	30
		M	6	2	56	19
	Gamma-Poisson	D	5	0	48	18
		M	8	0	132	32
Dynamic	Beta-Bernoulli	D	18	7	146	20
		M	11	4	116	40
	Gamma-Poisson	D	16	4	137	37
		M	10	3	70	28
		MultiSeg	8	0	59	9
		Reference	17	5	135	53

Table 6.6: Total number of identified topics by the joint model approaches.

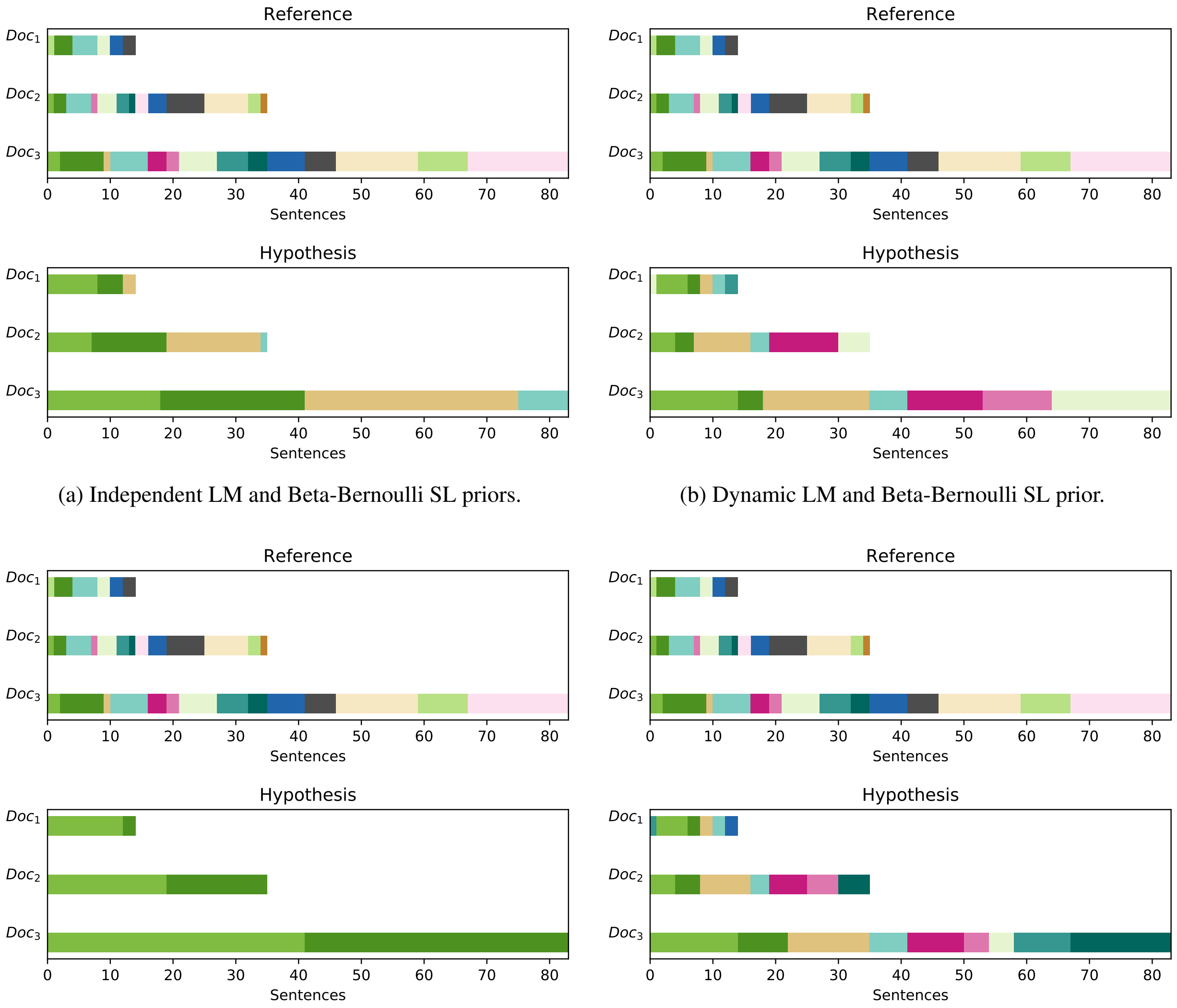
The conclusion from these experiments is that the proposed BeamSeg model is effective for topic identification in domains with prevalent topic development throughout the segments of the documents. This is the case of the AVL and Physics domains where BeamSeg obtains the best results out of all bench-marked algorithms. This is inline with the segmentation results and provides evidence that the tasks are related and should be modeled jointly since better results can be obtained. To achieve the best performance, it is necessary to use a combination a dynamic LM prior with a Gamma-Poisson SL prior at the modality level. Therefore, the proposed modeling assumptions fit well with the data. For the Biography, and News domains, the pipeline approach is more suitable. Our proposed graph-community detection algorithm obtains the best results for the News domain, and DBSCAN in the Biography domain.

6.3 Qualitative Analysis

We now perform a qualitative analysis of the topic identification results. We provide visual examples with different prior configuration to better understand the behavior of BeamSeg.

Figure 6.1 provides topic identification examples from biography documents of Amelia Earhart for the different BeamSeg prior configurations. One can observe that most of the time BeamSeg prefers to output topics that follow a similar order in all documents of the dataset. This assumption is explicitly made in dynamic LM prior but also occurs for the independent version since it actually matches the reference data. As mentioned before, in the Biography domain, the independent LM prior outputs few segments. The consequence for the topic identification task is that many of the topics are already merged because they belong to the same segment (Figures 6.1a and Figure 6.1c). When using a dynamic LM prior more segments emerge, making more room for cross-document topic identification (Figures 6.1b and Figure 6.1d). For example, the dark green segments in Figure 6.1a from D_2 and D_3 align the topics from the reference well but mix unrelated topics. As more segments are identified in the hypothesis, the topic alignments with the reference are maintained, and, consequently, more topics are correctly identified. This can be observed in Figure 6.1a by looking at the same utterance span as before. In this case, the light blue topic has an accurate match in the reference. This is the main difference in the quality of the topic identification between these prior configurations. When comparing the Gamma-Poisson and Beta-Bernoulli SL priors with the dynamic LM we see that the topic identification does not differ significantly, which is inline with the close B^3 scores. Therefore, the quality of the topic identification is related to the accuracy of the segmentation.

Figure 6.2 provides topic identification examples in four documents from the Physics dataset with the 'Frictional Forces' subject. The examples show the different possible prior configurations using the Gamma-Poisson since it obtains the best performance. When using the independent LM prior, the topic identification patterns are similar to the ones observed in the Biography domain. The identified segments tend to follow the same topic ordering (Figure 6.2a). Looking at the examples with a dynamic LM prior we can see that combining it with a modality SL prior affords more accurate results than the dataset-based SL prior. For example, the last segments of D_1 and D_3 are only correctly identified when using the modality prior. Another example is the last two segments of the reference from D_4. BeamSeg merges these segments both in the dataset and modality SL priors cases. The difference is that in the former this segment shares the same topic with another segment from D_3, whereas in the latter it is correctly identified as a local topic. Another observation we made while examining the different topic identification outputs is that the topic structure in the Physics and AVL domains have a higher complexity than in Jeong and Titov (2010)'s datasets. By this,



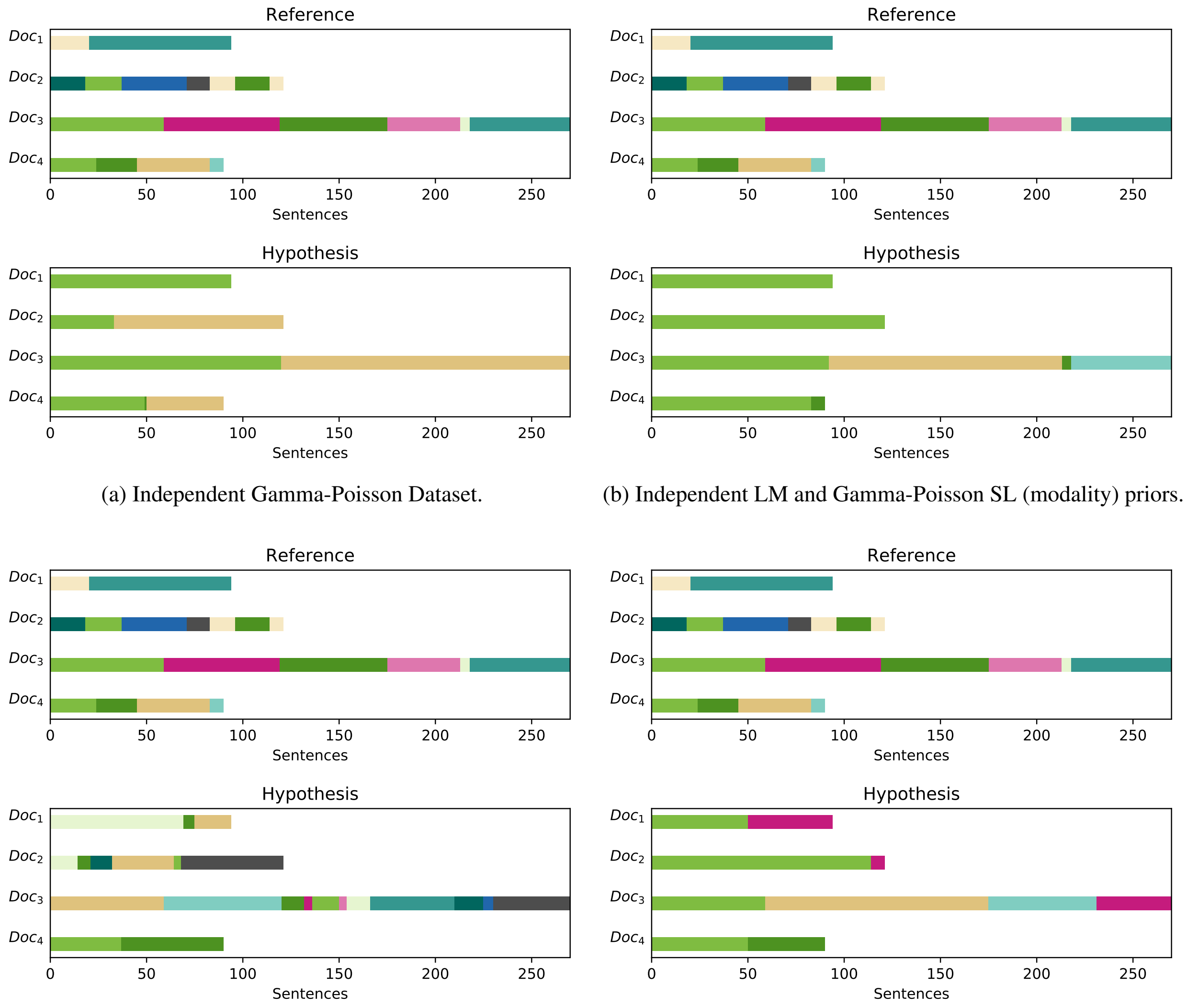
(a) Independent LM and Beta-Bernoulli SL priors.

(b) Dynamic LM and Beta-Bernoulli SL prior.

(c) Independent LM and Gamma-Poisson SL prior.

(d) Dynamic LM and Gamma-Poisson SL priors.

Figure 6.1: Topic identification examples in the Biography domain.



(a) Independent Gamma-Poisson Dataset.

(b) Independent LM and Gamma-Poisson SL (modality) priors.

(c) Dynamic LM and Gamma-Poisson SL (dataset) priors.

(d) Dynamic LM and Gamma-Poisson (modality) priors.

Figure 6.2: Topic identification examples in the Physics domain.

we mean that the topics sequences vary more across the different documents. This, coupled with documents that are harder to segment accurately and with our assumption that a shared topic order between documents exists, explains the worse results in the topic identification task for the AVL and Physics domains.

7 Conclusions

In this thesis, we studied how lexical-cohesion manifest in the context of a collection of documents describing similar topics, and hypothesized that its vocabulary relationships at the segment level could be used to improve the text segmentation and topic identification tasks in a joint and multi-document perspective. Given the multi-document scenario, we also hypothesize that segment length characteristics tied to document modality can be used to improve the results in these tasks. To test the previous hypothesis, it was necessary to acquire gold standard data since existing datasets do not allow us to study this phenomenon at a proper scale. To determine the extent of the advantages of a joint model we compared it with a pipeline approach. For the pipeline approach, we proposed two extensions to single-document models to the multi-document segmentation case (non-joint models) and a graph-community detection-based algorithm for topic identification. In this context, our main proposal is BeamSeg, an unsupervised Bayesian joint model for text segmentation and topic identification. By using a probabilistic approach, we can encode assumptions on how input data was generated. The novel modeling assumptions we encode in BeamSeg match the hypothesis we make in this thesis, and, thus, allows us to have a better grasp on how to answer our research questions. Given the previous research context, we summarize our contributions and point to future work in the following sections.

7.1 Contributions

Main contributions:

- We proposed BeamSeg, an unsupervised Bayesian joint model for segmentation and topic identification. BeamSeg is a mixture model where it is assumed that lexical cohesion can be observed across documents, meaning that segments describing the same topic use a similar lexical distribution over the vocabulary.

- We proposed an incremental MAP optimization procedure to carry out inference under the BeamSeg model that was shown to obtain adequate segmentations. We also explored a research line to apply Variational Inference instead, but we concluded that this is not feasible in our setup.

- We evaluated the proposed BeamSeg approach in available datasets and our collected learning materials. In the evaluation, we benchmarked several different state-of-the-art models, encompassing different types of modeling approaches to segmentation and topic identification, and compared their results to BeamSeg. The results show that BeamSeg obtained the best results, both in segmentation and topic identification, in AVL and Physics domains. These are domains where segments develop their corresponding topic more thoroughly, and, thus, more suitable to explore our target lexical cohesion phenomenon. The experimental setup also allowed to compare the impact of the proposed dynamic language model and segment length priors. From the results, we can see that only by using a dynamic language model and a modality based Gamma-Poisson segment length prior achieves the best results. This argues in favor of both our hypothesis, that is, a joint approach that models that vocabulary relationships between segments and segment length properties abstracted at the modality level can improve text segmentation and topic identification.

Secondary contributions:

- We carried out a data collection task to obtain learning materials from different domains and modalities. The dataset was manually annotated with segmentation boundaries and the corresponding topics identified. This enables research in multi-document segmentation and topic at a scale that was not possible before and in documents where segment topic development occurs more extensively. We also carried out a human agreement study for both segmentation and topic identification. For the topic identification task, this is, to the best of our knowledge, the first reported human agreement study. The results show that it is possible, to some extent, to observe agreement between human judges for both tasks. We found that the main cause for disagreement, are the different perceptions of what the level of granularity should be (Mota et al., 2018b). Through this data annotation study, we also found a phenomenon between segmentation and topic identification that had not been reported before. We found that it is possible that segments tightly couple two topics that can also be observed disjointly in other segments.

- We proposed an extension to the existing Bayesseg model for the multi-document case, Bayesseg-MD. The extended approach searches for similar utterances in the dataset and adds the corresponding word counts to segment likelihood estimations, making this a hybrid model with lexical similarity and probabilistic features (Mota et al., 2016). This approach obtained comparable results to BeamSeg in the AVL domain.

- We proposed an extension to the existing PLDA model for the multi-document case, PLDA-MD. This extension allowed us to study multi-document segmentation in a mixed-membership modeling perspective. From our Gibbs sampler derivations, we show that the multi-document aspect of PLDA-MD amounts to adding the counts from all topic assignments when sampling for new topic assignments. We take advantage of this setup to implement an algorithm that speeds up the Gibbs sampling procedure by caching the topic sampling probabilities. To obtain a significant number of cache hits, it is necessary to assume a fix scan order of the variables. This raises the question if the Gibbs sampler convergence is affected. In our experiments, we compared random and fixed scan orders. The results show that the convergence of the Gibbs sampler is similar in both cases, and, thus, we can use effectively use the caching algorithm. The results show that improvements from the PLDA version could be obtained in four out of five of the tested domains. For the AVL domain, the results are comparable to the best performances.

- We studied how a pipeline strategy for segmentation and topic identification compares to a joint model approach. In this context, we proposed an algorithm that given a segmentation assigns topic labels to segments based on the output of graph-community detection algorithms. When compared to a standard clustering algorithm, it was possible to obtain better results in three of the tested domains, making it a viable approach in a pipeline scenario for topic identification (Mota et al., 2018a).

7.2 Future Directions

Given the presented results and conclusions of this thesis, we highlight the following higher priority future directions:

- The rawest assumption we make in BeamSeg is that there is a shared topic ordering among all documents. Although this only happens to a certain extent in the datasets, it still allows us to make use of a dynamic language model prior, which can improve segmentation and topic identification. The results analysis indicates that many of the errors in these tasks stem from this assumption, and, thus, BeamSeg can certainly benefit if the model is improved in this aspect. One way we can at least relax this shared ordering assumption is by using the local/global topic mechanism from MultiSeg. This way we would only apply the dynamic language model prior to global topics.
- The inference procedure that we presented for BeamSeg is a MAP estimation. Ideally, we want to access the full posterior distribution during inference since it allows us to obtain more accurate results. A possible research line is to follow the inference approach used by Goldwater et al. (2009)

in a word segmentation task, a problem that infants must solve when acquiring a language. The inference procedure uses a Hidden Semi-Markov approach. In this setup, a blocked Gibbs sampler scheme is used, which means that a full utterance is randomly resampled instead of doing this at the word level. To sample new word segmentation boundaries a forward-backward procedure is used. The appeal in using this procedure is that it provides mixing efficiency by implicitly considering all possible segmentations of the given utterance at the same time.

- BeamSeg assumes that there is no overlap between different topic segments. Although it does not occur extensively, we do report that it is possible to find more than one topic in the same segments and also find them individually in other segments. To model this type of interaction one could use a mixed membership model. PLDA-MD does have such characteristics, and one possibility would be to incorporate them in BeamSeg.

- The segmentation granularity has been an issue in this research area. For a given document, defining segmentations with different granularity levels can make sense. To better study these differences, it is necessary to gather enough annotations for different segmentation patterns to emerge. The next step would be to organize the identified patterns in a hierarchical segment structure. This allows a more precise evaluation since we would be able to characterize segmentation models according to how well they match different possible paths in the hierarchy instead of a single segmentation reference.

Bibliography

G. Adelson-Velsky and E. M. Landis. An Algorithm for the Organization of Information. *Doklady Akademii Nauk USSR*, 146(2):263–266, 1962.

B. G. Ahn, B. Van Durme, and C. Callison-Burch. WikiTopics: What is popular on Wikipedia and why. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics, Workshop on Summarization*, pages 461–486, 2011.

Y.-Y. Ahn, J. Bagrow, and S. Lehmann. Link communities reveal multiscale complexity in networks. *Nature*, 466:761–4, 2010.

A. A. Alemi and P. Ginsparg. Text Segmentation based on Semantic Word Embeddings. *arXiv e-prints*, art. arXiv:1503.05543, 2015.

E. Amigó, J. Gonzalo, J. Artiles, and F. Verdejo. A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information Retrieval*, 12(4):461–486, 2009.

S. Arnold, R. Schneider, P. CudrÃ©-Mauroux, F. A. Gers, and A. LÃ¶ser. Sector: A neural model for coherent topic segmentation and classification. *Transactions of the Association for Computational Linguistics*, 7:169–184, 2019.

J. Atchison and S. Shen. Logistic-normal distributions:Some properties and uses. *Biometrika*, 67(2):261–272, 1980.

L. Backstrom, D. Huttenlocher, J. Kleinberg, and X. Lan. Group formation in large social networks: Membership, growth, and evolution. In *Proceedings of the International Conference on Knowledge Discovery and Data Mining*, pages 44–54, 2006.

A. Balagopalan, L. L. Balasubramanian, V. Balasubramanian, N. Chandrasekharan, and A. Damodar. Automatic keyphrase extraction and segmentation of video lectures. In *Proceedings of the International Conference on Transformations in Engineering Education*, pages 152–162, 2012.

D. Beeferman, A. Berger, and J. Lafferty. Statistical models for text segmentation. *Machine Learning*, 34: 177–210, 1999.

B. Bidyuk and R. Dechter. Cutset sampling with likelihood weighting. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, pages 39–46, 2006.

C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer-Verlag, New York, 2006.

D. Blei and J. Lafferty. Correlated Topic Models. In *Advances in Neural Information Processing Systems*, volume 18, pages 147–154. MIT Press, 2006a.

D. M. Blei and J. D. Lafferty. Dynamic topic models. In *Proceedings of the International Conference on Machine Learning*, pages 113–120, 2006b.

D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

V. Blondel, J. Guillaume, R. Lambiotte, and E. Mech. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, P10008:1–12, 2008.

A. Bougouin, F. Boudin, and B. Daille. Topicrank: Graph-based topic ranking for keyphrase extraction. In *Proceedings of the International Joint Conference on Natural Language Processing*, 2013.

S.-H. Chen and E. H. Ip. Behavior of the gibbs sampler when conditional distributions are potentially incompatible. *Journal of Statistical Computation and Simulation*, 85:1–10, 2014.

J. Chien and C. Lee. Deep unfolding for topic models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(2):318–331, 2018.

F. Y. Y. Choi. Advances in domain independent linear text segmentation. In *Proceedings of the North American Chapter of the Association for Computational Linguistics Conference*, pages 26–33, 2000.

A. Clauset, M. E. J. Newman, and C. Moore. Finding community structure in very large networks. *Physical Review E*, 70:066111, 2004a.

A. Clauset, M. E. J. Newman, and C. Moore. Finding community structure in very large networks. *Physical Review E*, pages 1–6, 2004b.

J. Cohen. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1):37, 1960.

L. J. Cronbach. Coefficient alpha and the internal structure of tests. *Psychometrika*, 16(3):297–334, 1951.

L. Du, W. L. Buntine, and M. Johnson. Topic segmentation with a structured topic model. In L. Vanderwende, H. D. III, and K. Kirchhoff, editors, *Proceedings of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 190–200, 2013.

J. Eisenstein. Hierarchical text segmentation from multi-scale lexical cohesion. In *Proceedings of North American Chapter of the Association for Computational Linguistics*, pages 353–361, 2009.

J. Eisenstein and R. Barzilay. Bayesian unsupervised topic segmentation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 334–343, 2008.

C. Fournier. Evaluating text segmentation using boundary edit distance. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 1702–1712, 2013.

W. N. Francis and H. Kucera. The Brown Corpus: A Standard Corpus of Present-Day Edited American English, 1979. Brown University Liguistics Department.

B. J. Frey and D. Dueck. Clustering by passing messages between data points. *Science*, 315(5814):972–976, 2007.

M. Galley, K. McKeown, E. Fosler-Lussier, and H. Jing. Discourse segmentation of multi-party conversation. In *Proceedings of the Annual Meeting on Association for Computational Linguistics*, pages 562–569, 2003.

M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002.

S. Goldwater, T. L. Griffiths, and M. Johnson. A bayesian framework for word segmentation: Exploring the effects of context. *Cognition*, 112:21–54, 2009.

T. L. Griffiths and M. Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101(suppl 1):5228–5235, 2004.

M. A. Halliday and R. Hasan. *Cohesion in English.* Longman, London, 1976.

B. D. He, C. D. Sa, I. Mitliagkas, and C. Ré. Scan order in gibbs sampling: Models in which it matters and bounds on how much. In *Annual Conference on Neural Information Processing Systems*, pages 1–9, 2016.

J. He, Z. Hu, T. Berg-Kirkpatrick, Y. Huang, and E. P. Xing. Efficient correlated topic modeling with topic embedding. In *Proceedings of the International Conference on Knowledge Discovery and Data Mining*, pages 225–233, 2017.

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

S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

M. D. Hoffman, D. M. Blei, and F. Bach. Online learning for latent dirichlet allocation. *Advances in Neural Information Processing Systems*, 23:856–864, 2010.

P. Hsueh, J. D. Moore, and S. Renals. Automatic segmentation of multiparty dialogue. In *Proceedings of the European Chapter of the Association for Computational Linguistics*, pages 9–10, 2006.

W. Huang. PhraseCTM: Correlated topic modeling on phrases within Markov random fields. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 521–526. Association for Computational Linguistics, 2018.

P. Jahnichen, F. Wenzel, M. Kloft, and S. Mandt. Scalable generalized dynamic topic models. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, Proceedings of Machine Learning Research, pages 1427–1435, 2018.

S. Jameel and W. Lam. An unsupervised topic segmentation model incorporating word order. In *Proceedings of the International Conference on Research and Development in Information Retrieval*, pages 203–212, 2013.

A. Janin, D. Baron, J. Edwards, D. Ellis, D. Gelbart, N. Morgan, B. Peskin, T. Pfau, E. Shriberg, A. Stolcke, and C. Wooters. The icsi meeting corpus. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, pages 364–367, 2003.

M. Jeong and I. Titov. Multi-document topic segmentation. In *Proceedings of the International Conference on Information and Knowledge Management*, pages 1119–1128, 2010.

M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37(2):183–233, Nov. 1999.

S. Joty, G. Carenini, and R. T. Ng. Topic segmentation and labeling in asynchronous conversations. *Journal of Artificial Intelligence Research*, 47(1):521–573, 2013.

A. Kazantseva and S. Szpakowicz. Linear text segmentation using affinity propagation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 284–293, 2011.

D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques*. Adaptive computation and machine learning. MIT Press, 2009.

O. Koshorek, A. Cohen, N. Mor, M. Rotman, and J. Berant. Text segmentation as a supervised learning task. In *Proceedings of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, volume 2 (Short Papers), pages 469–473, 2018.

K. Krippendorff. *Content Analysis: An Introduction to Its Methodology (second edition)*. Sage Publications, 2004.

J. Leskovec and J. J. Mcauley. Learning to discover social circles in ego networks. In *Advances in Neural Information Processing Systems 25*, pages 539–547. Curran Associates, Inc., 2012.

S. Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28(2):129–137, 2006.

O. Maimon and L. Rokach. *Data Mining and Knowledge Discovery Handbook*. Springer-Verlag, New York, 2005.

I. Malioutov and R. Barzilay. Minimum cut model for spoken lecture segmentation. In *Proceedings of the International Conference on Computational Linguistics and the Annual Meeting of the Association for Computational Linguistics*, pages 25–32, 2006.

T. P. Minka. Estimating a dirichlet distribution. Technical report, 2000.

P. Mota, M. Eskenazi, and L. Coheur. Multi-document topic segmentation using bayesian estimation. In *Proceedings of the International Workshop on Semantic Multimedia*, pages 443–447, 2016.

P. Mota, L. Coheur, and M. Eskénazi. Efficient navigation in learning materials: An empirical study on the linking process. In *Artificial Intelligence in Education*, pages 230–235, 2018a.

P. Mota, M. Eskénazi, and L. Coheur. MUSED: A multimedia multi-document dataset for topic segmentation. *Natural Language Engineering*, 24(6):921–946, 2018b.

M. E. J. Newman. Finding community structure in networks using the eigenvectors of matrices. *Physical Review E*, 74:036104, 2006.

V.-A. Nguyen, J. Boyd-Graber, and P. Resnik. SITS: A hierarchical nonparametric model using speaker identity for topic segmentation in multiparty conversations. In *ceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 78–87, 2012.

H. Noh, M. Jeong, S. Lee, J. Lee, and G. G. Lee. Script-description pair extraction from text documents of english as second language podcast. In *Proceedings of the International Conference on Computer Supported Education*, pages 5–10, 2010.

Y. Papanikolaou, J. R. Foulds, T. N. Rubin, and G. Tsoumakas. Dense distributions from sparse samples: Improved gibbs sampling parameter estimators for lda. *Journal of Machine Learning Research*, 18(1): 2058–2115, 2017.

R. J. Passonneau and D. J. Litman. Discourse segmentation by human and automated means. *Comput. Linguist.*, 23(1):103–139, 1997.

J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, volume 14, pages 1532–1543, 2014.

L. Pevzner and M. A. Hearst. A critique and improvement of an evaluation metric for text segmentation. *Computational Linguistics*, 28(1):19–36, 2002.

J. Pitman and M. Yor. The two-parameter Poisson-Dirichlet distribution derived from a stable subordinator. *Annals of Probability*, 25(2):855–900, 1997.

L. M. Pons, Pascal. Computing communities in large networks using random walks. *Journal of Graph Algorithms and Applications*, 10(2):191–218, 2006.

M. Purver, T. L. Griffiths, K. P. Körding, and J. B. Tenenbaum. Unsupervised topic modelling for multi-party spoken discourse. In *Proceedings of the International Conference on Computational Linguistics and the Annual Meeting of the Association for Computational Linguistics*, pages 17–24, 2006.

U. N. Raghavan, R. Albert, and S. Kumara. Near linear time algorithm to detect community structures in large-scale networks. *Physics Review E*, 76:036106, 2007.

R. Ranganath, S. Gerrish, and D. Blei. Black Box Variational Inference. In S. Kaski and J. Corander, editors, *Proceedings of the International Conference on Artificial Intelligence and Statistics*, pages 814–822, 2014.

M. Riedl and C. Biemann. Topictiling: A text segmentation algorithm based on lda. In *Proceedings of Association for Computational Linguistics Student Research Workshop*, pages 37–42, 2012.

C. P. Robert and G. Casella. *Monte Carlo Statistical Methods*. Springer-Verlag, Berlin, 2005.

T. Salimans and D. A. Knowles. Fixed-form variational posterior approximation through stochastic linear regression. *Bayesian Analysis*, 8(4):837–882, 2013.

J. Sander, M. Ester, H.-P. Kriegel, and X. Xu. Density-based clustering in spatial databases: The algorithm dbscan and its applications. *Data Mining and Knowledge Discovery*, 2(2):169–194, 1998.

L. K. Saul and M. I. Jordan. Exploiting tractable substructures in intractable networks. In *Proceedings of the International Conference on Neural Information Processing Systems*, pages 486–492, 1995.

W. A. Scott. Reliability of Content Analysis: The Case of Nominal Scale Coding. *Public Opinion Quarterly*, 19:321–325, 1955.

R. R. Shah, Y. Yu, A. D. Shaikh, and R. Zimmermann. TRACE: linguistic-based approach for automatic lecture video segmentation leveraging wikipedia texts. In *Proceedings of the International Symposium on Multimedia*, pages 217–220, 2015.

D. Shahaf, C. Guestrin, and E. Horvitz. Trains of thought: Generating information maps. In *Proceedings of the International Conference on World Wide Web*, 2012.

P. Shrout and J. Fleiss. Intraclass correlations: Uses in assessing rater reliability. *Psychological Bulletin*, 86 (2):420–428, 3 1979.

A. Srivastava and C. Sutton. Autoencoding Variational Inference For Topic Models. *arXiv e-prints*, art. arXiv:1703.01488, 2017.

B. Sun, P. Mitra, C. L. Giles, J. Yen, and H. Zha. Topic segmentation with shared topic detection and alignment of multiple documents. In *Proceedings of the International Conference on Research and Development in Information Retrieval*, pages 199–206, 2007.

Y. W. Teh, D. Newman, and M. Welling. A collapsed variational bayesian inference algorithm for latent dirichlet allocation. In *Advances in Neural Information Processing Systems*, pages 1353–1360, 2007.

A. Terenin, D. Simpson, and D. Draper. Asynchronous Gibbs Sampling. *arXiv e-prints*, 2015.

A. Terenin, S. Dong, and D. Draper. Gpu-accelerated gibbs sampling: a case study of the horseshoe probit model. *Statistics and Computing*, 29(2):301–310, 2019.

M. J. Wainwright and M. I. Jordan. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1(1-2):1–305, 2008.

L. Wang, S. Li, Y. Lv, and H. WANG. Learning to rank semantic coherence for topic segmentation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1340–1344, 2017.

N. G. Ward, S. D. Werner, D. G. Novick, E. E. Shriberg, C. Oertel, and T. Kawahara. The similar segments in social speech task, 2013.

S. Watanabe, T. Iwata, T. Hori, A. Sako, and Y. Ariki. Topic tracking language model for speech recognition. *Computer Speech Language*, 25(2):440–461, 2011.

Y. Weiss. Segmentation using eigenvectors: A unifying view. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 975–982, 1999.

J. Yang and J. Leskovec. Overlapping community detection at scale: A nonnegative matrix factorization approach. In *Proceedings of the International Conference on Web Search and Data Mining*, pages 587–596, 2013.

J. Yang, J. J. McAuley, and J. Leskovec. Community detection in networks with node attributes. In *Proceedings of the International Conference on Data Mining*, pages 1151–1156, 2013.

Q. Zhu, Z. Feng, and X. Li. GraphBTM: Graph enhanced autoencoded variational inference for biterm topic model. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 4663–4672, 2018.

D. Ziou and S. Tabbone. Edge detection techniques - an overview. *Pattern Recognition and Image Analysis*, 8:537–559, 1998.

Appendices

A Notation

In this Appendix, for convenience, we provide a summary of the variables and their corresponding description for the derivations in the proposed models. The adopted notation for Bayesseg (Section 4.2.1), PLDA-MD (Section 4.2.2), and BeamSeg (Section 4.4) are in Tables A.1, A.2, and A.3 respectively.

W	Number of words in the vocabulary.
U	Set of all utterance indexes.
u	Utterance index.
S	Segmentation of a document.
s	Segment of a document.
s_u	Segment assignment of utterance u.
$\mathbf{x}_u$	Bag-of-words vector of utterance u.
$\mathbf{X}$	The set of all utterances in the document collection.
ϕ	Per topic word probability distributions.
Φ	Set of all per topic word probability distributions.
β	Hyperparameter for the ϕ prior.
n_w^s	Number of times word w occurs in s.

Table A.1: Notation used in Bayesseg.

W	Number of words in the vocabulary.
w	Word index in the vocabulary.
u	Utterance index.
W_u	Number of words in sentence u.
$\mathbf{z}$	Topic assignments of words in all documents.
$\mathbf{c}$	Sentence segment boundary assignments in all documents.
ϕ	Per topic word probability distributions.
θ	Topic proportions of segments in all documents.
π	Topic segment boundary probability of documents.
β	Hyperparameter for the ϕ prior.
α	Hyperparameter for the θ prior.
γ	Hyperparameter for the π prior.
D	Number of documents in the collection.
U_d	Number of sentences in document d.
n_1^d	Number of sentences with $c = 1$ in document d.
n_0^d	Number of sentences with $c = 0$ in document d.
n_1^D	Number of sentences with $c = 1$ in document collection D.
n_0^D	Number of sentences with $c = 0$ in document collection D.
K	Number of topics.
$\phi_{k,w}$	Probability of word w in topic k.
$n_{D,w}^k$	Number of times word w was assigned topic k in D.
n_D^k	Number of words assigned to k in D.
S_u	Segment index containing u.
$U_{d,1}$	Sentence indexes in d with $c = 1$.
θ_{d,S_u}	Topic proportions of segment S_u from document d.
n_{d,S_u}^k	Number of words assigned to k in S_u from d.
$n_{d,S_u}^{\cdot}$	Total number of words in S_u from d.
$z_{d,w_{u,i}}$	Topic assignment of word $w_{u,i}$ from d.
$\mathbf{z}_{\neg(d,w_{u,i})}$	All word topic assignments except for word $w_{u,i}$ from d.
S_u^0	The S_u segment resulting from the merge at utterance u.
S_u^1	The S_u segment resulting from the split at utterance u.

Table A.2: Notation used in PLDA-MD.

W	Number of words in the vocabulary.
w	Word index in the vocabulary.
U	Set of all utterance indexes.
u	Utterance index.
K	Number of topics.
k	Topic index.
$\mathbf{x}_u$	Bag-of-words vector of utterance u.
$\mathbf{X}$	The set of all utterances in the document collection.
z_u	Topic assignment of utterance u.
$\mathbf{z}$	Topic assignments of all utterances in the document collection.
ϕ	Per topic word probability distributions.
Φ	Set of all per topic word probability distributions.
β	Hyperparameter for the ϕ prior.
$n_{U,w}^{k}$	Number of times word w is assigned topic k in all U utterances.
n_{U}^{k}	Number of times topic k occurs in all U utterances.
α_k	Precision parameter of topic k.
$\hat{\phi}_{k'w}$	Mean language model word probabilities of topic k.
γ	Variational parameters.
$\gamma_{d,i}^{k}$	Variational parameter of the i^{th} word in d for topic k.
$\gamma_{w}^{D_k}$	Sum of all k^{th} components of the $\gamma_{d,i}^{k}$ variational parameters for all words that match w.
D	Number of documents in the collection.
d	Document index.
i	i^{th} word in some document d.
$z_{d,i}$	Topic assignment of the i^{th} word in d.
$\mathbf{z}_{\neg d,i}$	All topic assignment except the i^{th} word in d.
$n_{w}^{D_k}$	Number of times word w is assigned topic k in D.
$n_{w}^{D_{k,\neg d,i}}$	Similar to the previous, but excludes from the counts the i^{th} word from document d.
$n_{\cdot}^{D_k}$	Number of times topic k appears in D.

Table A.3: Notation used in BeamSeg.

B PLDA-MD Derivations

We now work out the complete derivations of the Gibbs sampler for the PLDA-MD model. For convenience, we provide in Table A.3 a description of the used notation. The first step consists in deriving the joint probability expression of the model, since it is one of the terms of the Gibbs sampling equations. From the plate diagram in Figure 4.1 we get the following joint probability distribution for PLDA-MD:

$$\begin{aligned} p(\mathbf{w}, \mathbf{z}, \mathbf{c}, \Phi, \Theta, \pi, \beta, \alpha, \gamma) = & p(\mathbf{w}|\mathbf{z}, \phi)p(\Phi|\beta) \\ & p(\mathbf{z}|\Theta)p(\Theta|\mathbf{c}, \alpha) \\ & p(\mathbf{c}|\pi)p(\pi|\gamma) \end{aligned} \tag{B.1}$$

Sampling all latent variables is a computationally expensive step given the number of variables in PLDA-MD. To address this problem, we take advantage of conjugacy to integrate out[1] some of variables, building a collapsed Gibbs sampler. Conjugacy means that the posterior distribution is in the same family as the prior. In this context, we first define the topic shift probability of documents, $p(\pi|\gamma)$. These variables are drawn from a Beta prior, and, assuming γ symmetric parameters[2], its probability is defined as follows:

$$\begin{aligned} p(\pi|\gamma) &= \prod_{d=1}^{D} \frac{\Gamma(2\gamma)}{\Gamma(\gamma)^2} \pi_d^{\gamma-1}(1-\pi_d)^{\gamma-1} \\ &= \left(\frac{\Gamma(2\gamma)}{\Gamma(\gamma)^2}\right)^D \prod_{d=1}^{D} \pi_d^{\gamma-1}(1-\pi_d)^{\gamma-1}, \end{aligned} \tag{B.2}$$

where π_d is the topic shift probability of document d and D is the size of the dataset.

$p(\mathbf{c}|\pi)$ is the probability of the utterances being a segment boundary, given the corresponding document topic shift probability. Since these are Bernoulli distributed, we defined the expression according to:

$$p(\mathbf{c}|\pi) = \prod_{d=1}^{D} \pi_d^{n_1^d}(1-\pi_d)^{n_0^d}, \tag{B.3}$$

[1] A process also referred as marginalization in the literature.

[2] All parameters have the same value.

where n_1^d is the number of utterance boundaries in document d, and n_0^d is the number of non-utterance boundaries. Since both Equation B.2 and B.3 use π_d, we can join $p(\pi|\gamma)$ and $p(\mathbf{c}|\pi)$ in a single expression:

$$\begin{aligned} p(\mathbf{c}|\pi)p(\pi|\gamma) &= \left(\frac{\Gamma(2\gamma)}{\Gamma(\gamma)^2}\right)^D \prod_{d=1}^{D} \pi^{n_1^d}(1-\pi)^{n_0^d}\pi_d^{\gamma-1}(1-\pi_d)^{\gamma-1} \\ &= \left(\frac{\Gamma(2\gamma)}{\Gamma(\gamma)^2}\right)^D \prod_{d=1}^{D} (\pi^{n_1^d+\gamma-1})(1-\pi^{n_0^d+\gamma-1}) \qquad \text{(B.4)} \end{aligned}$$

From Equation B.4 we can actually see that the two distributions are conjugate, because what we obtained is also a beta distribution. The only thing missing is the normalizing constant $B = \frac{\Gamma(n_1^d+n_0^d+2\gamma)}{\Gamma(n_1^d+\gamma)\Gamma(n_0^d+\gamma)}$. Therefore, if we add the normalizing constant and integrate with respect to π, we simplify the expression, since integrating a distribution with respect to its parameters evaluates to one:

$$\begin{aligned} p(\mathbf{c}) &= \int p(\mathbf{c}|\pi)p(\pi|\gamma)d\pi \qquad \text{(B.5)} \\ &= \left(\frac{\Gamma(2\gamma)}{\Gamma(\gamma)^2}\right)^D \prod_{d=1}^{D} \frac{1}{B} \int B(\pi^{n_1^D+\gamma-1})(1-\pi^{n_0^D+\gamma-1})d\pi_d \\ &= \left(\frac{\Gamma(2\gamma)}{\Gamma(\gamma)^2}\right)^D \prod_{d=1}^{D} \frac{\Gamma(n_1^d+\gamma)\Gamma(n_0^d+\gamma)}{\Gamma(N_d+2\gamma)} \end{aligned}$$

where N_d is the number of segments in document d. Note how π does not appear in Equation B.5, meaning it is was integrated out.

Using an approach similar to the previous one, we can simplify the expression $p(\mathbf{w}|\mathbf{z},\Phi)p(\Phi|\beta)$. The first factor is the probability of generating words $\mathbf{w}$ given the topics assignments $\mathbf{z}$ and language models Φ. The second factor is the probability of the language models givens its prior β. Using the probability density function definition of a Dirichlet distribution,

$$\text{Dirichlet}(\beta) = \frac{\Gamma(\sum_{i=1}^{W}\beta_i)}{\prod_{i=1}^{W}\Gamma(\beta_i)}, \qquad \text{(B.6)}$$

we define each of the factors, assuming symmetric β parameters:

$$p(\Phi|\beta) = \left(\frac{\Gamma(\sum_{w=1}^{W}\beta)}{\prod_{w=1}^{W}\Gamma(\beta)}\right)^K \prod_{k=1}^{K}\prod_{w=1}^{W} \phi_{k,w}^{\beta-1} = \left(\frac{\Gamma(W\beta)}{\Gamma(\beta)^W}\right)^K \prod_{k=1}^{K}\prod_{w=1}^{W} \phi_{k,w}^{\beta-1} \qquad \text{(B.7)}$$

$$p(\mathbf{w}|\mathbf{z},\Phi) = \prod_{d=1}^{D}\prod_{k=1}^{K}\prod_{w=1}^{W}\phi_{k,w}^{n_{d,w}^{k}} = \prod_{k=1}^{K}\prod_{w=1}^{W}\phi_{k,w}^{n_{D,w}^{k}} \tag{B.8}$$

where K is number of topics, W the size of vocabulary, and $n_{d,w}^{k}$ the number of times word w is assigned topic k in document d. Similarly to before, we can note that Equations B.7 and B.8 are products over the same sets and the same $\phi_{k,w}$ variables. Thus, we join both factors as follows:

$$p(\mathbf{w}|\mathbf{z},\Phi)p(\Phi|\beta) = \left(\frac{\Gamma(W\beta)}{\Gamma(\beta)^{W}}\right)^{K}\prod_{k=1}^{K}\prod_{w=1}^{W}\phi_{k,w}^{n_{D,w}^{k}+\beta-1} \tag{B.9}$$

The multinomial and the Dirichlet are conjugate distributions, thus the resulting distribution in Equation B.9 is also Dirichlet. In this context, we can integrate out Φ using $C' = \left(\frac{\Gamma(W\beta)}{\Gamma(\beta)^{W}}\right)^{K}$, and add in the missing Dirichlet normalizing constant $D' = \frac{\Gamma(\sum_{w=1}^{W} n_{D,w}^{k}+\beta)}{\prod_{w=1}^{W}\Gamma(n_{D,w}^{k}+\beta)}$:

$$\begin{aligned}
p(\mathbf{w}|\mathbf{z}) &= \int p(\mathbf{w}|\mathbf{z},\Phi)p(\Phi|\beta)d\Phi \\
&= C'\prod_{k=1}^{K}\int\prod_{w=1}^{W}\phi_{k,w}^{n_{D,w}^{k}+\beta-1}d\phi_{k} \\
&= C\prod_{k=1}^{K}\frac{1}{D'}\int D'\phi_{k,w}^{n_{D,w}^{k}+\beta-1}d\phi_{k} \\
&= \left(\frac{\Gamma(W\beta)}{\Gamma(\beta)^{W}}\right)^{K}\prod_{k=1}^{K}\frac{\prod_{w=1}^{W}\Gamma(n_{D,w}^{k}+\beta)}{\Gamma(\sum_{w=1}^{W}n_{D,w}^{k}+W\beta)} \\
&= \left(\frac{\Gamma(W\beta)}{\Gamma(\beta)^{W}}\right)^{K}\prod_{k=1}^{K}\frac{\prod_{w=1}^{W}\Gamma(n_{D,w}^{k}+\beta)}{\Gamma(n_{D}^{k}+W\beta)}
\end{aligned}$$

The last factor we need to work out for the joint probability expression is $p(\mathbf{z}|\Theta)p(\Theta|\mathbf{c},\alpha)$, which models the interactions between the topic assignments $\mathbf{z}$, topic proportions Θ, and segmentation $\mathbf{c}$. These are defined with multinomial and Dirichlet distributions, similarly to the language models Φ. Thus, integrating out Θ follows the same pattern as before. The difference is that instead of counting the number of times a words occurs under a topic, we count the number of times a topic occurs in a segment. Assuming symmetric prior parameters α, we define each of the factors as follows:

$$p(\Theta|\mathbf{c},\alpha) = \prod_{d=1}^{D} \prod_{u \in U_{d,1}} \frac{\Gamma(K\alpha)}{\Gamma(\alpha)^K} \prod_{k=1}^{K} \theta_{d,S_u}^{\alpha-1} \tag{B.10}$$

$$p(\mathbf{z}|\Theta) = \prod_{d=1}^{D} \prod_{u \in U_{d,1}} \prod_{k=1}^{K} \theta_{d,S_u}^{n_k^{d,S_u}}, \tag{B.11}$$

where $U_{d,1}$ is the set of utterance indexes in d with $c_{d,u} = 1$, S_u is the segment index containing u, θ_{d,S_u} is the segment topic proportions of segment S_u in d, and n_k^{d,S_u} is the number of times topic k occurs in S_u. We can now integrate out Θ, with normalizing constants $C = \frac{\Gamma(K\alpha)}{\Gamma(\alpha)^K}$, and $D' = \frac{\Gamma(\sum_{k=1}^{K} n_{d,S_u}^{t} + \alpha)}{\prod_{k=1}^{K} \Gamma(n_{d,S_u}^{k} + \alpha)}$:

$$\begin{aligned}
p(\mathbf{z}|\mathbf{c}) &= \int p(\mathbf{z}|\Theta) p(\Theta|\mathbf{c},\alpha) d\Theta \\
&= C^{n_1^D} \prod_{d=1}^{D} \prod_{u \in U_{d,1}} \int \prod_{k=1}^{K} \theta_{d,S_u}^{n_{d,S_u}^{k} + \alpha - 1} d\theta_{d,S_u} \\
&= C^{n_1^D} \prod_{d=1}^{D} \prod_{u \in U_{d,1}} \frac{1}{D'} \int D' \prod_{k=1}^{K} \theta_{d,S_u}^{n_{d,S_u}^{k} + \alpha - 1} d\theta_{d,S_u} \\
&= C^{n_1^D} \prod_{d=1}^{D} \prod_{u \in U_{d,1}} \frac{\prod_{k=1}^{K} \Gamma(n_{d,S_u}^{k} + \alpha)}{\Gamma(n_{d,S_u}^{\cdot} + K\alpha)} \\
&= \left(\frac{\Gamma(K\alpha)}{\Gamma(\alpha)^T}\right)^{n_1^D} \prod_{d=1}^{D} \prod_{u \in U_{d,1}} \frac{\prod_{k=1}^{K} \Gamma(n_{d,S_u}^{k} + \alpha)}{\Gamma(n_{d,S_u}^{\cdot} + K\alpha)}
\end{aligned} \tag{B.12}$$

where n_1^D is the number of segments in the dataset D, and $n_{d,S_u}^{\cdot}$ the total number of words in segment S_u.

With the previous derivations we were able to get the expression for the joint distribution of the PLDA-MD model. The next step in building the Gibbs sampler is to define what the sampling equations are for each of the latent variables. After integrating out some of the latent variables, we are left with topic assignments $\mathbf{z}$ and segment boundaries $\mathbf{c}$. The Gibbs sampling equations works by sampling individual variables and the the corresponding equation is the probability of the sampled variable given all the other ones. Applying this principle to $\mathbf{z}$ we get the following sampling equation:

$$\begin{aligned} p(z_{d,w_{u,i}}|\mathbf{z}_{\neg(d,w_{u,i})},\mathbf{c},\mathbf{w}) &= \frac{p(z_{d,w_{u,i}},\mathbf{z}_{\neg(d,w_{u,i})},\mathbf{c},\mathbf{w})}{p(\mathbf{z}_{\neg(d,w_{u,i})},\mathbf{c},\mathbf{w})} \\ &= \frac{p(\mathbf{z},\mathbf{c},\mathbf{w})}{p(\mathbf{z}_{\neg(d,w_{u,i})},\mathbf{c},\mathbf{w})} \\ &= \frac{p(\mathbf{c})p(\mathbf{w}|\mathbf{z})p(\mathbf{z}|\mathbf{c})}{p(\mathbf{c})p(\mathbf{w}|\mathbf{z}_{\neg(d,w_{u,i})})p(\mathbf{z}_{\neg(d,w_{u,i})}|\mathbf{c})} \\ &= \frac{p(\mathbf{w}|\mathbf{z})p(\mathbf{z}|\mathbf{c})}{p(\mathbf{w}|\mathbf{z}_{\neg(d,w_{u,i})})p(\mathbf{z}_{\neg(d,w_{u,i})}|\mathbf{c})} \end{aligned} \tag{B.13}$$

where $w_{u,i}$ is the i^{th} word in utterance u. The previous derivation starts by applying the conditional probability definition. The result is an expression with the joint probability in the numerator, and joint distribution minus the sampled variable in the denominator. Then, we can note that $p(\mathbf{c})$ does not involve the variable we are excluding, $z_{d,w_{u,i}}$, and, thus, it appears both on the numerator and the denominator, canceling out. The derivation follows by working on similar terms in the numerator and the denominator individually. For $\frac{p(\mathbf{w}|\mathbf{z})}{p(\mathbf{w}|\mathbf{z}_{\neg(d,w_{u,i})})}$, the denominator and the numerator are very similar. Therefore, we expect to be able to cancel out many of the factors in the expression. For the remaining ones, we need to understand what are the implications of removing $z_{d,w_{u,i}}$ from the joint probability expression. These implications depend on what the current value of the variable is and what is the value we are sampling. For $\mathbf{z}$ we define the new topic value to be sampled as k'. Two scenarios are possible: k' is the same topic as the current $z_{d,w_{u,i}}$, or it is a different topic. We will first work out the case for which k' and $z_{d,w_{u,i}}$ are the same topic:

$$\begin{aligned} \frac{p(\mathbf{w}|\mathbf{z})}{p(\mathbf{w}|\mathbf{z}_{\neg(d,w_{u,i})})} &= \frac{\left(\frac{\Gamma(W\beta)}{\Gamma(\beta)^W}\right)^K \prod_{k=1}^K \frac{\prod_{i'=1}^{W_u}\Gamma(n^k_{D,w_{u,i'}}+\beta)}{\Gamma(n^k_D+W\beta)}}{\left(\frac{\Gamma(W\beta)}{\Gamma(\beta)^W}\right)^K \frac{\Gamma(n^{k'}_{D,w_{u,i}}+\beta-1)}{\Gamma(n^{k'}_D+W\beta-1)} \prod_{k=1,k\neq k'}^K \frac{\prod_{i'=1}^{W_u}\Gamma(n^k_{D,w_{u,i'}}+\beta)}{\Gamma(n^k_D+W\beta)}} \\ &= \frac{\left(\frac{\Gamma(W\beta)}{\Gamma(\beta)^W}\right)^K \frac{\Gamma(n^{k'}_{D,w_{u,i}}+\beta)}{\Gamma(n^k_D+W\beta)}}{\left(\frac{\Gamma(W\beta)}{\Gamma(\beta)^W}\right)^K \frac{\Gamma(n^{k'}_{D,w_{u,i}}+\beta-1)}{\Gamma(n^{k'}_D+W\beta-1)}} \\ &= \frac{\Gamma(n^k_{D,w_{u,i}}+\beta)\Gamma(n^{k'}_D+W\beta-1)}{\Gamma(n^{k'}_{D,w_{u,i}}+\beta-1)\Gamma(n^{k'}_D+W\beta)} \\ &= \frac{(n^{k'}_{D,w_{u,i}}+\beta-1)\Gamma(n^{k'}_{D,w_{u,i}}+\beta-1)\Gamma(n^{k'}_D+W\beta-1)}{(n^{k'}_D+W\beta-1)\Gamma(n^{k'}_{D,w_{u,i}}+\beta-1)\Gamma(n^{k'}_D+W\beta-1)} \\ &= \frac{n^{k'}_{D,w_{u,i}}+\beta-1}{n^{k'}_D+W\beta-1}, \end{aligned} \tag{B.14}$$

where W_u is the number of words in u, $n^{k'}_{d,w_{u,i}}$ is number of times word $w_{u,i}$ was assigned topic k' in d, and $n^{k'}_D$ the number of times topic k' appears in the dataset D. The previous derivation starts by separating, in the denominator, the k' counts from the product over K. This is because we need to remove the topic assignment $z_{d,w_{u,i}}$ from the counts, which explains why -1 appears on the factor regarding k'. Since all other factors appear in the numerator and in the denominator, they cancel out. The final step is to use $\Gamma(n) = (n-1)\Gamma(n-1)$ to make the factors with Γ cancel out. It should be noted that this approach to the derivations is going to occur in all derivations for the Gibbs sampling equation derivations. For simplicity, we will gloss over some of these details from now on.

The other sampling case, $z_{d,w_{u,i}}$ is currently assigned a topic different from k', is similar:

$$\begin{aligned}\frac{p(\mathbf{w}|\mathbf{z})}{p(\mathbf{w}|\mathbf{z}_{\neg(d,w_{u,i})})} &= \frac{\left(\frac{\Gamma(W\beta)}{\Gamma(\beta)^W}\right)^K \frac{\Gamma(n^{k'}_{D,w_{u,i}}+\beta+1)}{\Gamma(n^{k'}_D+W\beta)}}{\left(\frac{\Gamma(W\beta)}{\Gamma(\beta)^W}\right)^K \frac{\Gamma(n^{k'}_{D,w_{u,i}}+\beta)}{\Gamma(n^{k'}_D+W\beta-1)}} \\ &= \frac{\Gamma(n^{k'}_{D,w_{u,i}}+\beta+1)\Gamma(n^{k'}_D+W\beta-1)}{\Gamma(n^{k'}_{D,w_{u,i}}+\beta)\Gamma(n^{k'}_D+W\beta)} \\ &= \frac{(n^{k'}_{D,w_{u,i}}+\beta)\Gamma(n^{k'}_{D,w_{u,i}}+\beta)\Gamma(n^{k'}_D+W\beta-1)}{(n^{k'}_D+W\beta-1)\Gamma(n^{k'}_{D,w_{u,i}}+\beta)\Gamma(n^{k'}_D+W\beta-1)} \\ &= \frac{n^{k'}_{D,w_{u,i}}+\beta}{n^{k'}_D+W\beta-1},\end{aligned} \tag{B.15}$$

The difference for this second sampling case is that now we need to consider adding one to the topic counts for k' in the numerator since we are sampling for a different topic assignment than the one in the current state. Finally, we generalize the two cases in a single equation, using the Kronecker δ function:

$$\frac{p(\mathbf{w}|\mathbf{z})}{p(\mathbf{w}|\mathbf{z}_{\neg(d,w_{u,i})})} = \frac{n^{k'}_{D,w_{u,i}}+\beta-\delta(z_{d,w_{u,i}},k')}{n^{k'}_D+W\beta-1} \tag{B.16}$$

Now we will work on the other factor for the Gibbs sampling equation of $z_{d,w_{u,i}}$, $\frac{p(\mathbf{z}|\mathbf{c})}{p(\mathbf{z}_{\neg(d,w_{u,i})}|\mathbf{z})}$. First, we examine the case where the current topic assignment $z_{d,w_{u,i}}$ is equal to k'. The derivations are similar to the previous one, we just need to note that since we are excluding $z_{d,w_{u,i}}$ in the denominator the only factor that will remain from the product is the one regarding segment S_u:

$$\begin{aligned}
\frac{p(\mathbf{z}|\mathbf{c})}{p(\mathbf{z}_{\neg(d,w_{u,i})}|\mathbf{z})} &= \frac{\left(\frac{\Gamma(K\alpha)}{\Gamma(\alpha)^K}\right)^{n_1^D} \frac{\Gamma(n^{k'}_{d,S_u}+\alpha)}{\Gamma(n^{\cdot}_{d,S_u}+K\alpha)}}{\left(\frac{\Gamma(K\alpha)}{\Gamma(\alpha)^K}\right)^{n_1^D} \frac{\Gamma(n^{k'}_{d,S_u}+\alpha-1)}{\Gamma(n^{\cdot}_{d,S_u}+K\alpha-1)}} \\
&= \frac{\Gamma(n^{k'}_{d,S_u}+\alpha)\Gamma(n^{\cdot}_{d,S_u}+K\alpha-1)}{\Gamma(n^{\cdot}_{d,S_u}+K\alpha)\Gamma(n^{k'}_{d,S_u}+\alpha-1)} \\
&= \frac{(n^{k'}_{d,S_u}+\alpha-1)\Gamma(n^{k'}_{d,S_u}+\alpha-1)\Gamma(n^{\cdot}_{d,S_u}+K\alpha-1)}{(n^{\cdot}_{d,S_u}+K\alpha-1)\Gamma(n^{\cdot}_{d,S_u}+K\alpha-1)\Gamma(n^{k'}_{d,S_u}+\alpha-1)} \\
&= \frac{n^{k'}_{d,S_u}+\alpha-1}{n^{\cdot}_{d,S_u}+K\alpha-1}
\end{aligned} \tag{B.17}$$

The other case, $z_{d,w_{u,i}}$ is different from k', follows the same approach:

$$\begin{aligned}
\frac{p(\mathbf{z}|\mathbf{c})}{p(\mathbf{z}_{\neg(d,w_{u,i})}|\mathbf{z})} &= \frac{\left(\frac{\Gamma(K\alpha)}{\Gamma(\alpha)^K}\right)^{n_1^D} \frac{\Gamma(n^{k'}_{d,S_u}+\alpha+1)}{\Gamma(n^{\cdot}_{d,S_u}+K\alpha)}}{\left(\frac{\Gamma(K\alpha)}{\Gamma(\alpha)^K}\right)^{n_1^D} \frac{\Gamma(n^{k'}_{d,S_u}+\alpha)}{\Gamma(n^{\cdot}_{d,S_u}+K\alpha-1)}} \\
&= \frac{\Gamma(n^{k'}_{d,S_u}+\alpha+1)\Gamma(n^{\cdot}_{d,S_u}+K\alpha-1)}{\Gamma(n^{\cdot}_{d,S_u}+K\alpha)\Gamma(n^{k'}_{d,S_u}+\alpha)} \\
&= \frac{(n^{k'}_{d,S_u}+\alpha)\Gamma(n^{k'}_{d,S_u}+\alpha)\Gamma(n^{\cdot}_{d,S_u}+K\alpha-1)}{(n^{\cdot}_{d,S_u}+K\alpha-1)\Gamma(n^{\cdot}_{d,S_u}+K\alpha-1)\Gamma(n^{k'}_{d,S_u}+\alpha)} \\
&= \frac{n^{k'}_{d,S_u}+\alpha}{n^{\cdot}_{d,S_u}+K\alpha-1}
\end{aligned} \tag{B.18}$$

Generalizing the two cases:

$$\frac{p(\mathbf{z}|\mathbf{c})}{p(\mathbf{z}_{\neg(d,w_{u,i})}|\mathbf{z})} = \frac{n^{k'}_{d,S_u}+\alpha-\delta(z_{d,w_{u,i}},k')}{n^{\cdot}_{d,S_u}+K\alpha-1} \tag{B.19}$$

Putting Equations B.16 and B.19 we obtain the full Gibbs sampling expression for $\mathbf{z}$:

$$\begin{aligned}
p(z_{d,w_{u,i}}=k'|\mathbf{z}_{\neg(d,w_{u,i})},\mathbf{c},\mathbf{w}) =& \frac{n^{k'}_{D,w_{u,i}}+\beta-\delta(z_{d,w_{u,i}},k')}{n^{k'}_{D}+W\beta-1} \times \\
& \frac{n^{k'}_{d,S_u}+\alpha-\delta(z_{d,w_{u,i}},k')}{n^{\cdot}_{d,S_u}+K\alpha-1}
\end{aligned} \tag{B.20}$$

Having defined how to sample $\mathbf{z}$, we are left with the hidden variables $\mathbf{c}$. The general definition of the sampling equation is the following:

$$\begin{aligned}
p(c_{d,u}|\mathbf{c}_{\neg(d,u)},\mathbf{z},\mathbf{w}) &= \frac{p(c_{d,u},\mathbf{c}_{\neg(d,u)},\mathbf{z},\mathbf{w})}{p(\mathbf{c}_{\neg(d,u)},\mathbf{z},\mathbf{w})}\\
&= \frac{p(\mathbf{c},\mathbf{z},\mathbf{w})}{p(\mathbf{c}_{\neg(d,u)},\mathbf{z},\mathbf{w})}\\
&= \frac{p(\mathbf{c})p(\mathbf{w}|\mathbf{z})p(\mathbf{z}|\mathbf{c})}{p(\mathbf{c}_{\neg(d,u)})p(\mathbf{w}|\mathbf{z})p(\mathbf{z}|\mathbf{c}_{\neg(d,u)})}\\
&= \frac{p(\mathbf{c})p(\mathbf{z}|\mathbf{c})}{p(\mathbf{c}_{\neg(d,u)})p(\mathbf{z}|\mathbf{c}_{\neg(d,u)})} \qquad \text{(B.21)}
\end{aligned}$$

As before, we need to break down this expression according to the different sampling cases. We define the new value we are sampling as c'. If $c' = 0$ a merge occurs, and if $c' = 1$ a split occurs. Also, for each of these cases, we need to consider what the current value of $c_{d,u}$ is. We start with the scenario where $c' = 0$ (merge) and the current value of $c_{d,u}$ is also 0. In this context, for the factor $\frac{p(\mathbf{c})}{p(\mathbf{c}_{\neg(d,u)})}$, with the constant $C' = \left(\frac{\Gamma(2\gamma)}{\Gamma(\gamma)^2}\right)^D$, we obtain:

$$\begin{aligned}
\frac{p(\mathbf{c})}{p(\mathbf{c}_{\neg(d,u)})} &= \frac{C' \prod_{d=1}^{D} \frac{\Gamma(n_1^d+\gamma)\Gamma(n_0^d+\gamma)}{\Gamma(N_d+2\gamma)}}{C' \frac{\Gamma(n_1^d+\gamma)\Gamma(n_0^d+\gamma-1)}{\Gamma(N_d+2\gamma)} \prod_{d'=1,d'\neq d}^{D} \frac{\Gamma(n_1^{d'}+\gamma)\Gamma(n_0^{d'}+\gamma)}{\Gamma(N_{d'}+2\gamma)}}\\
&= \frac{\Gamma(n_0^d+\gamma)\Gamma(N_d+2\gamma-1)}{\Gamma(N_d+2\gamma)\Gamma(n_0^d+\gamma-1)}\\
&= \frac{(n_0^d+\gamma-1)\Gamma(n_0^d+\gamma-1)\Gamma(N_d+2\gamma-1)}{(N_d+2\gamma-1)\Gamma(N_d+2\gamma-1)\Gamma(n_0^d+\gamma-1)}\\
&= \frac{n_0^d+\gamma-1}{N_d+2\gamma-1} \qquad \text{(B.22)}
\end{aligned}$$

This previous derivation uses the same approach as before, but in this case we need to realize that by excluding $c_{d,u}$, the factor in the product over documents that remains is the one regarding d. Also, since $c_{d,u}$ has a value of 0, we subtract one from the n_0^d counts. The other case, $c_{d,u}$ is 1, is also very similar. The only difference is that a -1 does not appears in the numerator. This happens because now the segmentation changes with the merge of two segments. This was not the case before since we were sampling for the value that $c_{d,u}$ already has. Generalizing both cases:

$$p(c_{d,u}|\mathbf{c}_{\neg(d,u)},\mathbf{z},\mathbf{w}) = \frac{n_0^d+\gamma-\delta(c_{d,u},c')}{N_d+2\gamma-1} \qquad \text{(B.23)}$$

For the other part of the Gibbs sampling equation, $\frac{p(\mathbf{z}|\mathbf{c})}{p(\mathbf{z}|\mathbf{c}_{\neg(d,u)})}$, assuming $c_{d,u}$ is 0 and using $C' =$

$\left(\frac{\Gamma(K\alpha)}{\Gamma(\alpha)^K}\right)^{n_1^D}$ we get:

$$\frac{p(\mathbf{z}|\mathbf{c})}{p(\mathbf{z}|\mathbf{c}_{\neg(d,u)})} = \frac{C' \prod_{d=1}^{D} \prod_{u' \in U_{d,1}} \frac{\prod_{k=1}^{K} \Gamma(n^k_{d,S_{u'}} + \alpha)}{\Gamma(n^{\cdot}_{d,S_{u'}} + K\alpha)}}{C' \prod_{d=1}^{D} \prod_{u' \in U_{d,1} \backslash Su} \frac{\prod_{k=1}^{K} \Gamma(n^k_{d,S_{u'}} + \alpha)}{\Gamma(n^{\cdot}_{d,S_{u'}} + K\alpha)}}$$
$$= \frac{\prod_{k=1}^{K} \Gamma(n^k_{d,S^0_u} + \alpha)}{\Gamma(n^{\cdot}_{d,S^0_u} + K\alpha)} \qquad \text{(B.24)}$$

where S^0_u is the resulting segmentation when considering $c' = 0$, which matches the current segmentation. The previous simplification stems from excluding segment S_u in the denominator, making most of the factors to cancel all out. The other case, in which $c_{d,u}$ is a boundary, leads to the exact same expression. The difference is that an actually merge occurs, since $c_{d,u}$ is a boundary, but this merge matches S^0_u.

Now we will work on the sampling the segment split, $c' = 1$. The $\frac{p(\mathbf{c})}{p(\mathbf{c}_{\neg(d,u)})}$ factor is similar to Equation B.22, the only difference is that we use n_1^D instead of n_0^D, since we are evaluating the probability of getting 1 when sampling $c'_{d,u}$. The final result is:

$$\frac{p(\mathbf{c})}{p(\mathbf{c}_{\neg(d,u)})} = \frac{n_1^D + \gamma - \delta(c_{d,u}, c')}{D + 2\gamma - 1} \qquad \text{(B.25)}$$

The factor $\frac{p(\mathbf{z}|\mathbf{c})}{p(\mathbf{z}|\mathbf{c}_{\neg(d,u)})}$, assuming $c_{d,u}$ is 1, results in the following expression:

$$\frac{p(\mathbf{z}|\mathbf{c})}{p(\mathbf{z}|\mathbf{c}_{\neg(d,u)})} = \frac{\left(\frac{\Gamma(K\alpha)}{\Gamma(\alpha)^K}\right)^{n_1^D} \prod_{d=1}^{D} \prod_{u' \in U_{d,1}} \frac{\prod_{k=1}^{K} \Gamma(n^k_{d,S_{u'}} + \alpha)}{\Gamma(n^{\cdot}_{d,S_{u'}} + K\alpha)}}{\left(\frac{\Gamma(K\alpha)}{\Gamma(\alpha)^K}\right)^{n_1^D - 1} \prod_{d=1}^{D} \prod_{u' \in U_{d,1} \backslash S_u, S_{u-1}} \frac{\prod_{k=1}^{K} \Gamma(n^k_{d,S_{u'}} + \alpha)}{\Gamma(n^{\cdot}_{d,S_{u'}} + K\alpha)}}$$
$$= \left(\frac{\Gamma(K\alpha)}{\Gamma(\alpha)^K}\right) \frac{\prod_{k=1}^{K} \Gamma(n^k_{d,S^1_{u-1}} + \alpha)}{\Gamma(n^{\cdot}_{d,S^1_{u-1}} + K\alpha)} \frac{\prod_{k=1}^{K} \Gamma(n^k_{d,S^1_u} + \alpha)}{\Gamma(n^{\cdot}_{d,S^1_u} + K\alpha)} \qquad \text{(B.26)}$$

where S^1_u is the resulting segmentation when considering $c' = 1$. Since we are dealing with a split case, removing $c_{d,u}$ affects two segments, S_u and S_{u-1}, and, thus, we need to exclude their topic assignment counts. This is why we end up with an extra fraction when compared to the merge case. When considering the case $c_{d,u}$ not being a boundary, the exact same thing happens. Therefore, we can use Equation B.26 in both situations.

Having finished all derivations for all factors and their possible sampling cases, we can put together the final Gibbs sampling equation for $\mathbf{c}$:

$$p(c_{d,u}=c'|\mathbf{c}_{\neg(d,u)},\mathbf{z},\mathbf{w})=\begin{cases}\frac{n_0^d+\gamma-\delta(c_{d,u},c')}{D+2\gamma-1}\frac{\prod_{k=1}^{K}\Gamma(n^k_{d,S_u^0}+\alpha)}{\Gamma(n^{\cdot}_{d,S_u^0}+K\alpha)}, c'=0\\ \frac{n_1^d+\gamma-\delta(c_{d,u},c')}{D+2\gamma-1}\frac{\Gamma(K\alpha)}{\Gamma(\alpha)^K}\frac{\prod_{k=1}^{K}\Gamma(n^k_{d,S_{u-1}^1}+\alpha)}{\Gamma(n^{\cdot}_{d,S_{u-1}^1}+K\alpha)}\frac{\prod_{k=1}^{K}\Gamma(n^k_{d,S_u^1}+\alpha)}{\Gamma(n^{\cdot}_{d,S_u^1}+K\alpha)}, c'=1\end{cases}$$

C Document Segmentation Annotation Instructions

Segmentation Study Instructions

Introduction

The goal of this research is to develop a system that helps students to browse learning materials efficiently. For example, when students are preparing for a physics exam and want to review the topic of kinetic energy they do not want to watch the full 1.5 hours video lecture. In the ideal case, students would only watch the portion of the lecture where this topic was covered. To this end, segmenting the lecture into individual topics is the first step in building such systems.

Topic Segmentation Guidelines

The annotation task you will perform consists of partitioning a set of learning materials (video lectures, HTML pages, Power Points, or PDFs) in a sequence of segments. This means that you need to identify places in the learning materials where a topic change occurs. After the identification of these boundaries, a segment is then defined by the set of sentences between a pair of sequential boundaries.

The guidelines for the task are as follows:

1. Every segment needs to be cohesive and self-contained.
> **Explanation:** only mark a boundary when the topic change contributes to the understanding of the content organization of the document.

2. Determine if the annotated segment can be easily assigned a small description regarding the corresponding topic. Some examples of descriptions are “gravity”, “centripetal acceleration”, or “a proof of Theorem A”
> **Explanation:** assigning a topic description helps to make sure that the segments are really self-contained and cohesive. If you are having difficulties coming up with a topic description to distinguish it from previous segments, then the segment may be a continuation of the previous segment.

3. It is not mandatory that visual text markers, such as bold titles for sections/subsections , match segment boundaries. You are free to combine or break the document at these markers if it is your understanding that this should be done.
> **Explanation:** it is possible that a new section on a document starts but the topic segment continues.

4. Sentences referring a common concept can be used with different goals, thus, they might belong to different segments. Each segment is defined to convey a clear topic or goal, thus, it is necessary to ensure that the context in which a sentence is used matches the overall goal of the corresponding segment.
> **Explanation:** concept sharing does not indicate that the sentences must be in the same segment. For example, a document might have a segment detailing the concept of *velocity* and later use *velocity* in another segment about *acceleration*.

Document Segmentation Practical Example

Below we provide annotation examples of a document. The example is part of a video lecture transcript about Newton's Laws. The green strings denote expected document segments and the red string corresponds to unexpected segments.

Example:

==========

You now know that you have to overcome inertia to get your bicycle moving, but what is it that allows you to overcome it?
Well, the answer is explained by Newton's Second Law.
In mathematical terms, Newton's Second Law says that force is the product of mass times acceleration.
To cause an object to accelerate, or speed up, a force must be applied.
The more force you apply, the quicker you accelerate.
And the more mass your bicycle has, and the more mass you have too, the more force you have to use to accelerate at the same rate.
This is why it would be really difficult to pedal a 10,000 pound bicycle.
And it is this force, which is applied by your legs pushing down on the pedals, that allows you to overcome Newton's Law of Inertia.
The harder you push down on the pedals, the bigger the force and the quicker you accelerate.

==========

Another question is why objects go forward when they start to move?
According to Newton's Third Law, for every action, there is an equal and opposite reaction.
To understand this, think about what happens when you drop a bouncy ball.
As the bouncy ball hits the floor, it causes a downward force on the floor.
This is the action.
The floor reacts by pushing on the ball with the same force, but in the opposite direction, upward, causing it to bounce back up to you.
Together, the floor and the ball form what's called the action/reaction pair.
Now when it comes to your bicycle, it is a little more complicated.
As your bicycle wheels spin clockwise, the parts of each tire touching the ground push backwards against the Earth: the actions.
The ground pushes forward with the same force against each of your tires: the reactions.
Since you have two bicycle tires, each one forms an action/reaction pair with the ground.
And since the Earth is really, really, really big compared to your bicycle, it barely moves from the force caused by your bicycle tires pushing backwards, but you are propelled forward.

==========

Explanation: the previous segmentation is acceptable because it reflects the two main topics of the document: Newton's 2nd Law and Newton's 3rd Law. Each of the segments clearly focuses on these topics, thus, they are cohesive and self-contained.

Counter example:

==========

You now know that you have to overcome inertia to get your bicycle moving, but what is it that allows you to overcome it?
Well, the answer is explained by Newton's Second Law.
In mathematical terms, Newton's Second Law says that force is the product of mass times

acceleration.
To cause an object to accelerate, or speed up, a force must be applied.
The more force you apply, the quicker you accelerate.
And the more mass your bicycle has, and the more mass you have too, the more force you have to use to accelerate at the same rate.
This is why it would be really difficult to pedal a 10,000 pound bicycle.
And it is this force, which is applied by your legs pushing down on the pedals, that allows you to overcome Newton's Law of Inertia.
The harder you push down on the pedals, the bigger the force and the quicker you accelerate.
==========
Another question is why objects go forward when they start to move?
According to Newton's Third Law, for every action, there is an equal and opposite reaction.
To understand this, think about what happens when you drop a bouncy ball.
As the bouncy ball hits the floor, it causes a downward force on the floor.
This is the action.
The floor reacts by pushing on the ball with the same force, but in the opposite direction, upward, causing it to bounce back up to you.
Together, the floor and the ball form what's called the action/reaction pair.
==========
Now when it comes to your bicycle, it is a little more complicated.
As your bicycle wheels spin clockwise, the parts of each tire touching the ground push backwards against the Earth: the actions.
The ground pushes forward with the same force against each of your tires: the reactions.
Since you have two bicycle tires, each one forms an action/reaction pair with the ground.
And since the Earth is really, really, really big compared to your bicycle, it barely moves from the force caused by your bicycle tires pushing backwards, but you are propelled forward.
==========

Explanation: the previous segmentation is not desirable because it is too fine grained. In the red segment case the annotator considered as a main topic applying Newton's 3rd Law to pedaling a bicycle. It should be considered that the goal of the bicycle example is to make the understanding of Newton's 3rd Law easier, thus, it can be aggregated with the previous segment.

Counter example:

==========
You now know that you have to overcome inertia to get your bicycle moving, but what is it that allows you to overcome it?
Well, the answer is explained by Newton's Second Law.
In mathematical terms, Newton's Second Law says that force is the product of mass times acceleration.
To cause an object to accelerate, or speed up, a force must be applied.
The more force you apply, the quicker you accelerate.
And the more mass your bicycle has, and the more mass you have too, the more force you have to use to accelerate at the same rate.
This is why it would be really difficult to pedal a 10,000 pound bicycle.
And it is this force, which is applied by your legs pushing down on the pedals, that allows you to overcome Newton's Law of Inertia.
The harder you push down on the pedals, the bigger the force and the quicker you accelerate.
Another question is why objects go forward when they start to move?
According to Newton's Third Law, for every action, there is an equal and opposite reaction.
To understand this, think about what happens when you drop a bouncy ball.

As the bouncy ball hits the floor, it causes a downward force on the floor.
This is the action.
The floor reacts by pushing on the ball with the same force, but in the opposite direction, upward, causing it to bounce back up to you.
Together, the floor and the ball form what's called the action/reaction pair.
Now when it comes to your bicycle, it is a little more complicated.
As your bicycle wheels spin clockwise, the parts of each tire touching the ground push backwards against the Earth: the actions.
The ground pushes forward with the same force against each of your tires: the reactions.
Since you have two bicycle tires, each one forms an action/reaction pair with the ground.
And since the Earth is really, really, really big compared to your bicycle, it barely moves from the force caused by your bicycle tires pushing backwards, but you are propelled forward.
==========

Explanation: the previous segmentation is not desirable because it is too coarse. The annotator might be thinking in Newton's Laws in a more general sense. When reading the document it is possible to observe that the two segments are strong enough to exist by themselves, meaning that they have different goals.

Task instructions:

- Annotate each file in the *$DIR* folder with document segment boundaries. The boundaries of the topic segments are represented by the string "==========". Please use this exact sequence of characters. You can copy this sequence from the beginning or the end of the document files.

- For each annotated segment boundary, indicate your level of certainty in the annotation. Use the character "C" if you are sure it is a correct segment boundary and the character "U" if you are uncertain. Finally, add a topic description of the segment as described in *guideline 2*.

- Open the link in the first line of each file in *$DIR* folder. The link contains the original format of the document (video, HTML page, Power Point, or PDF). When performing the annotation task the documents should be analyzed in their original format and the annotations done in the corresponding text file of the *$DIR* folder.

Using the previous practical example, a possible annotation is the following:

========== C Newton's 2nd Law
You now know that you have to overcome inertia to get your bicycle moving, but what is it that allows you to overcome it?
Well, the answer is explained by Newton's Second Law.
In mathematical terms, Newton's Second Law says that force is the product of mass times acceleration.
To cause an object to accelerate, or speed up, a force must be applied.
The more force you apply, the quicker you accelerate.
And the more mass your bicycle has, and the more mass you have too, the more force you have to use to accelerate at the same rate.
This is why it would be really difficult to pedal a 10,000 pound bicycle.
And it is this force, which is applied by your legs pushing down on the pedals, that allows you to overcome Newton's Law of Inertia.
The harder you push down on the pedals, the bigger the force and the quicker you accelerate.
========== C Newton's 3rd Law
Another question is why objects go forward when they start to move?

According to Newton's Third Law, for every action, there is an equal and opposite reaction.
To understand this, think about what happens when you drop a bouncy ball.
As the bouncy ball hits the floor, it causes a downward force on the floor.
This is the action.
The floor reacts by pushing on the ball with the same force, but in the opposite direction, upward, causing it to bounce back up to you.
Together, the floor and the ball form what's called the action/reaction pair.
========== U 3rd Law applied to bike pedaling
Now when it comes to your bicycle, it is a little more complicated.
As your bicycle wheels spin clockwise, the parts of each tire touching the ground push backwards against the Earth: the actions.
The ground pushes forward with the same force against each of your tires: the reactions.
Since you have two bicycle tires, each one forms an action/reaction pair with the ground.
And since the Earth is really, really, really big compared to your bicycle, it barely moves from the force caused by your bicycle tires pushing backwards, but you are propelled forward.
==========

D Topic Identification Annotation Instructions

Segment Relationship Identification Study Instructions

Introduction

The goal of this research is to develop a system that helps students to browse learning materials efficiently. For example, students preparing for a physics exam might want to find where the topic of kinetic energy is explained in different documents. The students do not want to skim all documents until they find this topic. In the ideal case, students would have the documents segmented by topic and similar segments from different documents grouped together. To this end, automatically identifying which segments are similar across different documents is a crucial step in building such systems.

Segment Relationship Identification Guidelines

The annotation task you will perform consists of identifying segments with similar content in a set of learning materials (video lectures, HTML pages, Power Points, or PDFs) . The segmentation of the documents is not part of this annotation task, it is given *a priori*. You only need to annotate which segments are similar to one another.

The guidelines for the task are as follows:

1. Segments sharing a common main topic should be annotated as being related.

> **Explanation**: it is not expected that segments in a relationship have the exact same semantic meaning. It is sufficient that they share a common main topic. For example, 2 segments can have the same main topic but one covers it in much more detail than the other.

2. It is sufficient that segments share a single main topic to annotate them as related.

> **Explanation:** segments can have more than one main topic.

3. Determine if the annotated segment relationship can be easily assigned a small description regarding the corresponding topic. Some examples of descriptions are “gravity”, “centripetal acceleration”, or “a proof of Theorem A”.

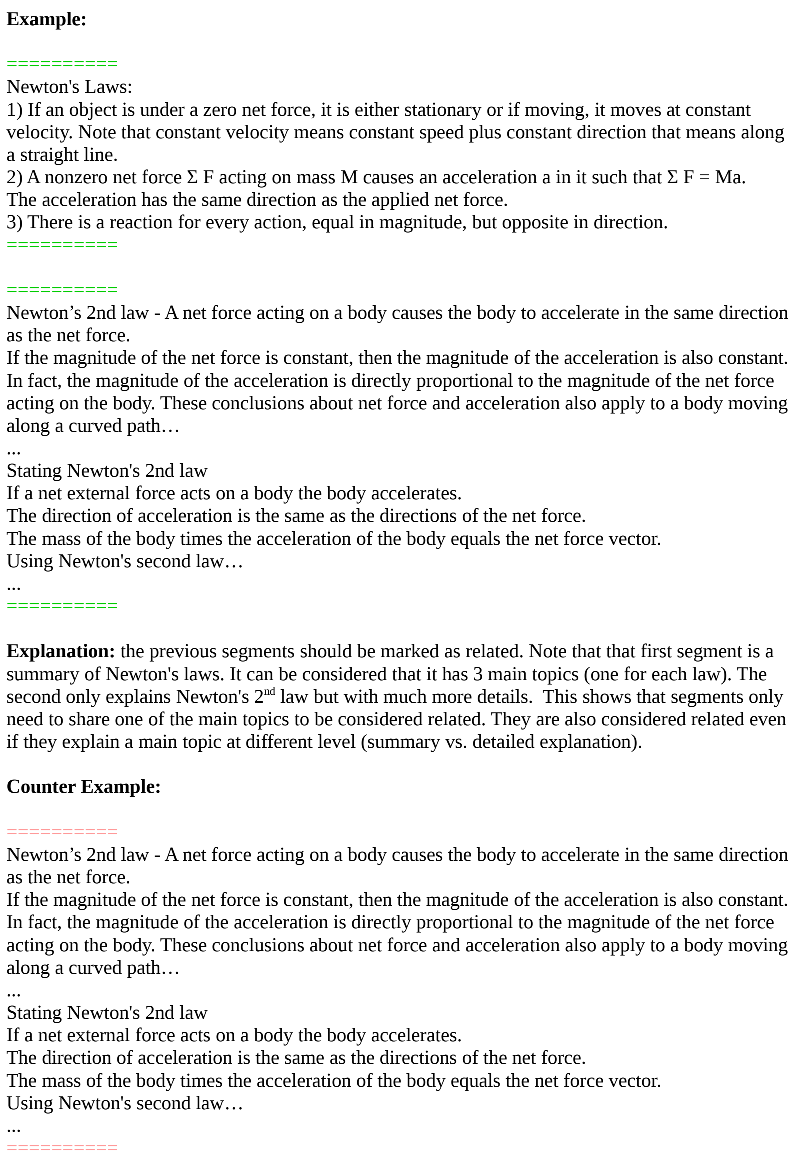
> **Explanation**: assigning a topic description helps to group segments that share the same main topic. If you are having difficulties coming up with a topic description which is different from existing descriptions, then the segments probably belongs to an existing group.

4. Segments may not have any relationship with any other segment.

> **Explanation:** some segments have topics that only relate with their corresponding document. For example, course related information such as homework deadlines.

Segment Relationship Identification Practical Example

Below we provide annotation examples. The examples are from documents about Newton's Laws. The green strings denote related segments and the red strings correspond to unrelated segments.



Example:

==========

Newton's Laws:
1) If an object is under a zero net force, it is either stationary or if moving, it moves at constant velocity. Note that constant velocity means constant speed plus constant direction that means along a straight line.
2) A nonzero net force Σ F acting on mass M causes an acceleration a in it such that Σ F = Ma. The acceleration has the same direction as the applied net force.
3) There is a reaction for every action, equal in magnitude, but opposite in direction.

==========

==========

Newton’s 2nd law - A net force acting on a body causes the body to accelerate in the same direction as the net force.
If the magnitude of the net force is constant, then the magnitude of the acceleration is also constant. In fact, the magnitude of the acceleration is directly proportional to the magnitude of the net force acting on the body. These conclusions about net force and acceleration also apply to a body moving along a curved path…
...
Stating Newton's 2nd law
If a net external force acts on a body the body accelerates.
The direction of acceleration is the same as the directions of the net force.
The mass of the body times the acceleration of the body equals the net force vector.
Using Newton's second law…
...

==========

Explanation: the previous segments should be marked as related. Note that that first segment is a summary of Newton's laws. It can be considered that it has 3 main topics (one for each law). The second only explains Newton's 2nd law but with much more details. This shows that segments only need to share one of the main topics to be considered related. They are also considered related even if they explain a main topic at different level (summary vs. detailed explanation).

Counter Example:

==========

Newton’s 2nd law - A net force acting on a body causes the body to accelerate in the same direction as the net force.
If the magnitude of the net force is constant, then the magnitude of the acceleration is also constant. In fact, the magnitude of the acceleration is directly proportional to the magnitude of the net force acting on the body. These conclusions about net force and acceleration also apply to a body moving along a curved path…
...
Stating Newton's 2nd law
If a net external force acts on a body the body accelerates.
The direction of acceleration is the same as the directions of the net force.
The mass of the body times the acceleration of the body equals the net force vector.
Using Newton's second law…
...

==========

==========
Newton's First Law: Consider a body on which no net force acts.
If the body is at rest, it will remain at rest.
If the body is moving with a constant velocity, it will continue to do so.
Example: Consider a cart on the air track.
The cart is floating on the track with no friction.
The net force on the cart is zero.
• If the cart is at rest, it remains stationary.
• If I give the cart a push it will move with constant velocity.
P Newton's Law is valid only in an inertial reference frame, a frame that is not accelerating, e.g. a powerless spacecraft far away from all planets (good example) or close to the surface of the Earth (good approximation).
Any frame that is moving at constant velocity with respect to an inertial frame is also an inertial frame.
==========

Explanation: the previous segments are not related. Despite both describe Newton's Laws, the former specifically describes Newton's 1st law, whereas the later specifically describes Newton's 2nd law, thus, they have different main topics.

Task instructions:

- In the provided zip file, each of the directories starting with the letter *L* contains a document for which segment relationships must be annotated. In each document directory, you will find a **_full_doc.txt* file containing the full textual representation of the document. You will also find **_segX.txt* files containing the individual segments of the documents. To annotate that 2 segments are related, copy the corresponding files to a common directory under the *seg_relations_annotation* directory. Note that you need to create these directories. For example, for the first segment of the first document you have to create a directory and copy the corresponding file to it. Later, in another document, if you find a related segment you just need to copy it to the previously created directory. The name of the directories should follow guideline **3**. That is, some possible directory names would be *gravity* or *centripetal_accelaration*.

- We strongly suggest that the annotation of the segments relationships of a document is done sequentially. When you annotate a document you must start in the first segment and annotate all the following ones. This means you cannot annotate segments from one document and move to other segments in other documents before finishing the current one. This ensures that the context in which a segment occurs is taken into account.

- Open the link in the first line of each **_full_doc.txt* file. The link contains the original format of the document (video, HTML page, Power Point, or PDF). When performing the annotation task the documents should be analyzed in their original format.