NLP Methods for Weak Signals Detection from Unstructured Text

Dissertation submitted in fulfilment of the requirements for the degree of PhD in Management

by Judicael POUMAY

Under the supervision of Ashwin ITTOO

Thesis Jury Members :

- Ashwin Ittoo (HEC Liège, Management School of the University of Liège)
- Pierre Geurts (Faculty of Applied Sciences, University of Liège)
- Michael Schyns (HEC Liège, Management School of the University of Liège)
- Antal van den Bosch (Utrecht University)
- Shankar Venkatagiri (Indian Institute of Management Bangalore)
- Le Minh Nguyen (Japan Advanced Institute of Science and Technology)

Contents

ABSTRACT

The exponential growth of unstructured and unlabeled data has created a pressing need for effective methods of extracting insights from such data. Natural language processing (NLP) has emerged as a powerful tool for analyzing and understanding human language in various real-world applications. However, weak signal and emerging trends detection, which involves processing large amounts of text data, has been limited by the use of predefined keywords or relatively simple topic modeling methods. In this work, we propose a hierarchical and temporal clustering method that enables the extraction of fine-grained information and the detection of smaller weak signals. Moreover, our method provides a way to understand the interaction and evolution of weak signals/emerging trends through topic correlations and temporal tracking. Furthermore, we introduce a novel approach to evaluate the accuracy, robustness, and flexibility of our method. Overall, this research offers a new and advanced method for weak signal detection through NLP, which can help organizations make informed decisions and stay ahead in today's dynamic market.

ACKNOWLEDGEMENTS

This dissertation would not have been possible without the support of many people. First and foremost, I owe a debt of gratitude to my supervisor, Ashwin Ittoo, whose guidance, trust, and encouragement have been instrumental to the successful completion of this project. As a master student, Ashwin gave me opportunities to hone my skills with multiple industry projects and continued to support me with the freedom and trust I needed to complete this thesis. He has been present whenever I needed, but never interfered, allowing me to lead this project as an independent researcher.

I am also grateful to my thesis committee comprised of Michael Schyns, Pierre Geurts, and Antal van den Bosch, who have provided me with valuable feedback and support over the years, steering this research in the best direction. I would also like to extend this gratitude to the external jury members, Shankar Venkatagiri and Le Minh Nguyen. Moreover, I would like to thank Michael Ghilissen who kindly took the time to provide insights and feedback on my manuscript.

My gratitude also extends to HEC-Liège and ULiège, where I have been a student for over nine years. They have provided me with the resources I needed to attend conferences across the world, expand my network, and conduct my research. In particular, I have been fortunate to experience HEC-Liège as a place filled with wonderful people and an environment that allows PhD students to develop as independent researchers.

I am grateful to the HEC-Liège Digital Lab, and in particular Nicolas Neysen, for the opportunity to work on this project in tandem with KPMG. I would also like to highlight KPMG and its team for their support and feedback, and for showing their trust in me and science through freedom and independence in research, in no particular order : Arnaud Hemricourt, Jean-François Villeret, Sven Muehlenbrock, Alexis Palm, Raphaël Schair, Pascal Denis, Axel Hristodoulakis, and Markus Lamest.

Finally, I am deeply grateful to my friends and family who have always believed in me throughout this endeavor. My parents, sister, and brother have always given me the tools to learn and grow, and I would not be the curious and diligent mind I am today without their unwavering support. My friends have also been a constant source of encouragement, and have helped me to become a better person.

Part I

INTRODUCTION

In today's rapidly changing and highly competitive global landscape, organizations must continually adapt to new challenges and seize opportunities to remain relevant. A crucial aspect of achieving this goal is that of environmental scanning, which refers to the continuous process of gathering, analyzing, and interpreting information from the external environment to pinpoint significant developments that may impact an organization [Glenn and Gordon, 2009]. In this thesis, we will be focusing on a sub-task of environmental scanning. Namely, detecting weak signals [Ansoff, 1975, Glenn and Gordon, 2009].

In this thesis, we define a weak signal as : "An ambiguous and potentially transformative cue, often conveyed through textual or communicative means, indicating an event, phenomenon, or subtle shift within a specific domain of interest. These signals are inherently uncertain and easy to dismiss, making their full impact and implications unclear. While seemingly insignificant in isolation, weak signals may possess latent potential to herald significant emergent trends or discontinuities, influencing the course of developments within the observed context." We will see how this definition came about in Chapter 1.

The exponential growth in data production in recent years has generated an abundance of unstructured and unlabeled information, encompassing scientific articles, press articles, blog posts, social media posts, and reports. This data is presumed to harbor valuable insights that can be extracted automatically to perform environmental scanning and weak signal detection [Hiltunen, 2007a]. However, this task can be challenging, as traditional manual analysis often struggles to effectively work on a large amount of unstructured and unlabeled data. Consequently, there is a pressing need for advanced techniques capable of mining insights from this diverse pool of information [Mühlroth and Grottke, 2018].

Natural Language Processing (NLP) is a rapidly evolving field of study that focuses on the interaction between computers and human languages [Cambria and White, 2014]. NLP systems employ a combination of machine learning, linguistics, and computer science to enable machines to understand, analyze, and generate human language. Owing to advances in deep learning algorithms, the field of NLP has witnessed significant progress in recent years [Zhao et al., 2023], culminating in a plethora of practical applications across various domains. Some notable examples include chatbots [Biswas, 2023] capable of engaging in natural conversations with users, machine translation systems [Stahlberg, 2020] that automatically translate text between languages, and sentiment analysis [Birjali et al., 2021] systems that determine the emotional tone of written text.

In particular, NLP can be used to detect weak signals [Mühlroth and Grottke, 2018]. By utilizing NLP techniques, it is possible to process automatically vast amounts of unstructured and unlabeled text data within a specific domain, enabling the identification of patterns and content related to weak signals. This approach has the potential to surpass traditional methods that rely on manual labor, as it facilitates the discovery of new weak signals that may be unknown or unanticipated. Ultimately, NLP holds the potential to revolutionize the way organizations approach strategic decision-making by offering early insights into potentially emerging trends and potential opportunities or threats.

Prior NLP research on weak signal detection has proven to be limited in their effectiveness, as they rely on basic keywords and statistical techniques to pinpoint infrequent and growing terms as potential weak signals [Yoon, 2012, Park and Cho, 2017, Lee and Park, 2018, Kim et al., 2016]. These approaches necessitate the predefinition of keywords, which hinders the discovery of new weak signals which, by their very nature, are unknown. To tackle this limitation, some researchers have investigated flat text clustering methods, such as topic models, which adopt a more exploratory approach and facilitate the discovery of novel weak signals [Kim and Lee, 2017, El Akrouchi et al., 2021, Breitzman and Thomas, 2015]. However, these topic modeling techniques come with their own set of limitations, as they solely extract large, high-level generic clusters [Paisley et al., 2015]. Thus, such techniques are not capable of capturing more subtle details present within the data such as weak signals. Furthermore, these methods necessitate the number of topics to be defined upfront. However, this value is unknown and difficult to estimate [Teh et al., 2006]. Since we do not know what we are supposed to extract, determining the appropriate number of topics to extract in advance becomes problematic.

In light of the limitations of previous NLP research on weak signal detection, our methodology aims to develop a more effective approach for identifying and tracking weak signals over time, as well as analyzing topic interactions for scenario planning and decision-making [Coffman, 1997]. The innovative methodology we propose combines a hierarchical and temporal topic model with topic tracking and correlations, enabling the extraction of both high-level and fine-grained topics and events in the data thus producing a topic hierarchy without the need to predefine the number of topics. Moreover, we also studied methods of Coreference Resolution, the task of clustering textual mention referring to the same reality, which could help in resolving duplicate mentions of events extracted. This scalable and unsupervised approach is designed to work with any kind of corpus from scientific articles to news articles, making it especially valuable for discovering unknown weak signals in any domain.

With this approach, small sub-topics (that may be localized in time) which cannot be traced back to previous periods are indicators of new emerging weak signals. In other words, the weakness of a signal (small sub-topics) and its novelty (cannot be traced back in time) are the main factors which allows us to detect weak signals. The topic correlation module in itself does not help us with the detection of weak signals but rather provide additional information about how these weak signals interact with other weak signals and larger trends. Thus, providing a platform for scenario building and strategic planning.

Ensuring the quality of the extracted topics is crucial in order to trust the insights they provide. However, evaluating unsupervised models, especially hierarchical ones, has proven to be a challenging task [Chang et al., 2009, Hoyle et al., 2021, Doogan and Buntine, 2021, Bhatia et al., 2017]. Numerous methods have been proposed for evaluating topic models, such as perplexity, topic coherence, and the Word Intrusion task. Unfortunately, these methods exhibit several shortcomings. For instance, perplexity [Blei et al., 2003] and topic coherence [Newman et al., 2010] have been shown to be uncorrelated with human judgment. The Word Intrusion task has been proposed as a solution [Chang et al., 2009, Lau et al., 2014] but we have demonstrated in Chapter 8 that it might be unreliable for evaluating hierarchical topic models.

To address these challenges, we propose a novel evaluation method for hierarchical topic models that leverages labeled datasets. This approach focuses on the ability to extract known topics, providing a more comprehensive understanding of the model's performance. By analyzing the document topic distribution and its ability to predict the actual labels of the documents, we can obtain a quantitative assessment of the model's quality, termed "label accuracy." While labelled dataset may not be easily available, this new evaluation method offers a promising new approach to assessing the quality and completeness of hierarchical topic models.

From the foregoing discussions, the research questions addressed by this thesis are :

- Q1 : Event & Entity Coreference Resolution (ECR)
  - Can we use ECR to support weak signal detection?
  - Is there a trade-off between performance (predictive & run-time) and embedding size?
  - How do the embeddings' performance compare within and across families?
  - Current state of the art methods for ECR relies on neural network and word embeddings methods. Hence, these research questions are focused on understanding the relationship between embedding size, performance (both predictive and run-time), and the comparison of performance among different embedding families in the context of Coreference Resolution (CR). The overall goal is to evaluate whether the current state of the art in ECR is sufficiently precise to support weak signal detection.
- Q2 : Hierarchical & Temporal Topic Modelling
  - How can hierarchical and temporal information be effectively integrated into topic models?
  - What are the challenges associated with combining hierarchical and temporal information in topic modeling?
  - Can we efficiently use Gibbs sampling and how does it compare to stochastic variational inference (SVI) in terms of speed?
  - These research questions focus on the integration of hierarchical and temporal information in topic modeling, the challenges associated with this integration, the evaluation and comparison of the proposed HTMOT method, and the performance of the novel implementation of Gibbs sampling.
- Q3 : Topic Tracking
  - Can semantic information be effectively used for topic tracking and how does it compare to lexical information?
  - What are the advantages and challenges of using semantic-based approaches for topic tracking compared to lexical-based approaches?
  - How does the proposed Semantic Divergence (SD) method, based on word embeddings, perform in topic tracking compared to existing methods?
  - What are the challenges of topic tracking in the context of hierarchical topic modelling?
  - These research questions revolve around investigating the use of semantic information for topic tracking, comparing it to lexical information, evaluating the proposed Semantic Divergence (SD) method, and understanding the challenges of topic tracking in hierarchical topic modelling.

- Q4 : Evaluating Topics & Topic Models
  - Can the current state of the art for topic model evaluation effectively evaluate hierarchical topic models, and what are its limitations?
  - How does the automated version of this method compare to the human version in evaluating hierarchical topic models?
  - Can we define a new methodology for evaluation and how does ot compare to a traditional evaluation method coherence?
  - Can the document topic distribution accurately predict the known labels of the documents, and what is the label accuracy achieved by the models?
  - How does the coherence of the taxonomy produced from the known labels reflect the quality and coherence of the hierarchical topic models?
  - Do the results of the experiments support the effectiveness of hierarchical topic models in extracting small sub-topics?
  - These research questions focus on evaluating and comparing different evaluation methods for hierarchical topic models, including the Word Intrusion task, label accuracy, completeness, unexpectedness, and coherence. The research also aims to investigate the performance of hierarchical topic models in extracting small sub-topics.

In this thesis, we will propose several novel methods for answering these research questions. The structure is as follows. The background (part 2) will define important concepts and provide a review of the related literature. It is divided into the following Chapters:

1. Trends and Weak Signals
2. Event & Entity Coreference Resolution
3. Word Embeddings
4. Topic models

Our methodology (part 3) is divided into the following Chapters:

1. Event & Entity Coreference Resolution
   - This Chapter was published as a long paper in EMNLP 2021 [1] with title "A Comprehensive Comparison of Word Embeddings in Event & Entity Coreference Resolution".
   - It is concerned with the research question Q1
2. Hierarchical Topic Modelling Over Time
   - This Chapter was submitted as a long paper in RANLP 2023 [2].
   - It is concerned with the research question Q2
3. Computing Topic Correlations
   - It is concerned with the research question Q2
4. Topic Tracking Using Word Embeddings

[1] https://2021.emnlp.org/

[2] http://ranlp.org/ranlp2023/

    - This Chapter was published as a long paper in NLDB 2022 [3] "Using Meaning Instead of Words to Track Topics".
    - It is concerned with the research question Q3

Our evaluation strategy (part 4) is divided into the following Chapters:

1. A Critic of the Current State Of The Art in Topic Model Evaluation for Hierarchical Topic Model
    - This Chapter was submitted as a long paper in ACL 2023 [4].
    - It is concerned with the research question Q4
2. A new Method for Topic Model Evaluation
    - This Chapter was submitted as a short paper in RANLP 2023[5].
    - It is concerned with the research question Q4

Finally, Section 5 concludes the thesis with a summary of the findings, limitations, and future research directions.

[3] https://link.springer.com/conference/nldb

[4] https://2023.aclweb.org/

[5] http://ranlp.org/ranlp2023/

Part II

BACKGROUND

The following Chapters will provide the necessary background for the rest of the thesis. We will start by presenting the concepts related to weak signals and previous work on the subject since it is the main goal of this thesis (see Chapter 1). In Chapter 2 and 3, we will follow with a literature review on Coreference Resolution and Word Embeddings which represent the first part of our methodology, specifically Chapter 5 about Word Embeddings in Coreference Resolution. Word Embeddings will also be used in Chapter 7 about Topic Tracking. Finally, Chapter 4 we will provide a background on topic models and related tasks which represent the core of our work. This will be important for Chapter 6 which is about our novel method for Hierarchical and Temporal Topic Modelling method, Chapter 7 which is about Topic Tracking, Chapter 8 which is about the Word Intrusion Task, and Chapter 9 which is about our novel evaluation method based on a labelled dataset.

1 Trends and weak signals

1.1 Introduction

Environmental scanning refers to the continuous process of gathering, analyzing, and interpreting information from the external environment to pinpoint significant developments that may impact an organization [Glenn and Gordon, 2009]. The information gathered often take the form of trends or weak signals observed in the firm's environment [Ansoff, 1975].

A trend is a general direction in which something is developing or changing. In the context of markets, trends are patterns or tendencies that emerge over a period of time. They can be long-term or short-term, upward or downward, and they can relate to economic, social, or technological factors. Trends are often used in forecasting and analysis to help predict future behavior or events based on historical patterns [akhavanhariri et al., 2022]. Trends can be divided into macro, micro, and emerging trends. Each of these categories serves different purposes and offers unique insights that can help businesses, policy makers, and individuals make informed decisions [Peloso, 2020].

Macro trends are broad and long-lasting trends that affect various sectors and demographics worldwide. They are driven by significant shifts in cultural, technological, or economic landscapes. Examples include the rise of artificial intelligence, climate change, or demographic shifts such as aging populations. Recognizing macro trends is crucial for strategic planning, as they shape the context in which organizations operate. They can influence business models, product development, and marketing strategies over the long term [Peloso, 2020].

Micro trends, on the other hand, are more localized and short-term in nature. They are specific to a particular industry, market segment, or geographic area. These trends often emerge from shifts in consumer preferences or advances in technology within a specific field. For instance, a new fashion style gaining popularity among teenagers or a new software tool being adopted by graphic designers would be considered micro trends. Micro trends provide opportunities for innovation and differentiation, especially for businesses targeting niche markets or looking for short-term growth opportunities [Peloso, 2020].

Emerging trends represent the early signals of change that have the potential to become significant micro and macro trends. They are often seen as the 'next big thing' but are currently in their infancy and not yet mainstream. For example, cryptocurrency or virtual reality in their early stages were considered emerging trends before they evolved into mainstream phenomena. Tracking emerging trends is important for businesses and individuals aiming to stay ahead of the curve, as they can highlight new market opportunities or potential disruptions [Kim et al., 2013].

Finally, weak signals are the earliest indicators of a potential emerging trend. Contrary to trends, they manifest more often as punctual events instead of a continuous trend. These signals are subtle, ambiguous, and easy to dismiss, but they can indicate significant changes. Weak signals could include a novel use of technology, a new behavior among a small group of consumers, or an unconventional business strategy that's beginning to show signs of success. Recognizing and interpreting weak signals can be challenging, but it's a crucial part of trend forecasting and strategic planning [Hiltunen, 2007b].

It is important to note that there is often interplay between these types of trends. A macro trend can be subdivided into multiple micro and emerging trends, and an emerging trend can grow to become a macro trend of its own. Conversely, multiple subsequent weak signals may eventually lead to the birth of a new emerging trend. Understanding this dynamic can provide a comprehensive view of the evolving landscape, enabling more effective planning and decision-making [Coffman, 1997].

1.2 A Bit of History on Weak Signals

The concept of weak signals was first proposed by Ansoff in 1975 [Ansoff, 1975] as a way to address the limitations of traditional strategic planning. He defined weak signals as incomplete and difficult-to-interpret indicators of potential future developments which could provide early warning of strategic discontinuities and allow firms to anticipate and prepare for them effectively. Hence, understanding weak signals enables firms to anticipate and prepare for potential threats or opportunities to stay ahead of the curve and gain a competitive advantage.

Hence, weak signals are subtle, emerging indicators that may provide early warning of potential future trends or events. They are part of the early lifecycle of trends [Andrus, 1997]; specifically, they are punctual events that indicate that a new trend might emerge. However, weak signals are often difficult to detect and interpret, particularly because they are often found close to the noise level of the global signal [Coffman, 1997]. Despite this, they can provide valuable insights into what the future may hold, making them an important tool for firms to stay ahead of the curve and gain a competitive advantage.

Weak signals may be found in various sources, such as social media, news articles, and expert interviews [Hiltunen, 2007a], and they can be related to a wide range of topics such as Political, Economic, Sociological, Technological, Legal, and Environmental changes (PESTEL) [Coffman, 1997]. They have a number of potential applications in various fields, such as risk management, strategic planning, and policy-making [Yoon, 2012]. By providing early warning of potential future developments, weak signals can help organizations and decision-makers anticipate and prepare for potential changes or challenges.

While weak signals are a valuable tool for anticipating potential future trends and events, they also create a paradox for planners. Waiting too long to act may result in missed opportunities while acting too quickly based on weak signals can lead to false alarms or misinterpretations. Therefore, it is important to recognize that weak signals are not always accurate or reliable and should be carefully evaluated and contextualized to be used effectively. Firms must exercise caution and discernment in interpreting weak signals and using them to inform strategic decision-making.

Weak signals are not naturally understood due to the current paradigm and their inherent weakness. Preparing for weak signals requires stepping out of our comfort zone and being creative. In particular, scenario planning is an effective method for studying potential outcomes based on weak signals. By making up stories about what the future may bring, we can infer what is possible and better anticipate discontinuities [Coffman, 1997].

For example, in 2004, Blockbuster was at its prime since its founding in 1985. They had become the de facto name in video rental and the company's fortunes seemed like they'd go on forever, at least internally. In 1997, Netflix was founded as an online DVD rental and streaming video startup. Blockbuster launched a competing DVD-by-mail service in 2004, which ultimately floundered, and the company subsequently declared bankruptcy in 2010[1]. This story exemplifies how ignoring emerging market disruption can lead to tragic consequences.

Creating a scenario requires a deep understanding of the multiple inter-dependencies of current and emerging technologies. One important aspect to consider is the auto-catalytic nature of weak signal ecosystems, where weak signals often depend on each other for their growth. For example, cars depend on engines and petroleum products to function, and petroleum products depend on engines for their distribution, while engines depend on cars and petroleum products. Understanding these inter-dependencies is crucial for identifying and interpreting weak signals. This is why weak signals are better found in uncertain environments since their chaotic potential attracts innovation [Coffman, 1997].

1.3 A Modern Understanding of Weak Signals

In the 2000s, Hiltunen [Hiltunen, 2007b] noticed that weak signals had not been analyzed formally in mainstream science and identified a significant debate about weak signals in Finland. She aimed to draw attention to this discussion on the international level and observed that there was a lack of coherence in the comprehension of weak signals, with various synonyms and closely related concepts being used and ambiguous features being identified.

Hiltunen drew on the field of semiotics, as pioneered by Peirce [Peirce, 1868], to formalize the concept of weak signals with her triadic model of future signs [Hiltunen, 2008]. The model consists of three dimensions (the issue, signal, and interpretation) which will be explained in the following paragraphs. To illustrate the triadic model, we can consider the emergence of self-landing rockets as a future sign, signaling a new era in space travel.

[1] `https://thefuturemarket.com/futurechronicle/the-value-of-weak-signals`

The issue refers to the underlying reality that a future sign represents and measures its magnitude. For example, with self-landing rockets, one measure of magnitude could be the number of times rockets have successfully self-landed, or more generally, the number of events related to this topic. The issue is at the core of future signs since it represents the actual reality that gives rise to them. Without an issue, no signal can be emitted, and no interpretation can be proposed [Kuusi and Hiltunen, 2012, Hiltunen, 2008].

The signal refers to the amount of discussion around the issue and includes any messages transmitted about it. These messages might be anthropogenic (created by someone) [Kuusi and Hiltunen, 2012], such as news articles, social media posts, scientific papers, or patents [Hiltunen, 2007b], or naturogenic [Kuusi and Hiltunen, 2012], meaning that the future sign itself has some direct effect on the observable world, which can be reported through anthropogenic signals. In most cases, people have access to the signal, but not to the issue. For example, one may not have witnessed a rocket self-landing but may have heard or read many reports about it. The signal is the surface of future signs; the visible part that people can perceive.

Lastly, *the interpretation* measures our perception of the disruptive/transformative potential of the future sign. It reflects whether people are hyped, invested, and hopeful or dismissive and pessimistic. Specifically, interpretation should be low when the potential impact is unclear, and high when its importance is clear. This is the only subjective component of the triadic model. In the case of self-landing rockets, it is now clear that this technology will revolutionize the space industry, but it was not always evident. Interpretation is the landscape that colors future signs, and it has two main sources: the interpretation of the reporter and the interpretation of the readers.

All three dimensions of the triadic model (issue, signal, and interpretation) are closely connected [Hiltunen, 2008, Kuusi and Hiltunen, 2012], as illustrated by the signification process described in [Kuusi and Hiltunen, 2012]. The issue is often latent, as it may not be directly observable, but it generates the signal. The signal is accessible and carries information about the issue, which can be interpreted by reporters and readers alike. Their interpretation (opinions and perceptions) about the potential impact of the future sign can color the facts presented in the signal.

Using the triadic model, we can distinguish between weak and strong signals. A weak signal is one that is weak in at least two of the three dimensions [Hiltunen, 2008]. For instance, in the early days of self-landing rockets, only a few attempts were successful (low issue), the media did not cover the topic extensively (low signal), and there was little enthusiasm or belief in the idea (low interpretation). In such cases, all three dimensions were small, indicating a weak signal.

Another example of a weak signal that later became a significant trend is the development of chatbots such as chatGPT. This issue dimension is represented by the creation and advancements of different chatbots, starting with Eliza and Cleverbot in 1996 and 1997. At the time, these chatbots represented a weak emerging issue compared to the current capabilities of chatGPT. The signal dimension represented by the news coverage of the issue remained mostly anecdotal until chatGPT. While Siri and Alexa were introduced more recently, they did not fundamentally disrupt the industry. However, ChatGPT has clearly disrupted the conversation, as evidenced by the increasing number of news articles discussing it. The interpretation dimension represented by the general opinion on the issue remained low and neutral as it was mostly of academic interest until chatGPT emerged. Now, many people believe that such chatbots will be extremely important in the coming years. Hence, we can observe how chatbots used to be a weak signal that grew over time to a strong signal.

Henceforth, we define a weak signal as : "An ambiguous and potentially transformative cue, often conveyed through textual or communicative means, indicating an event, phenomenon, or subtle shift within a specific domain of interest. These signals are inherently uncertain and easy to dismiss, making their full impact and implications unclear. While seemingly insignificant in isolation, weak signals may possess latent potential to herald significant emergent trends or discontinuities, influencing the course of developments within the observed context."

1.4 Detecting Weak Signals

The concept of weak signals has been primarily studied in the field of futures studies and management, but it is gaining attention from other communities, especially natural language processing (NLP). According to the systematic review by Muhlroth, between 2006 and 2016 only one to three articles are published per year on the subject of weak signal detection through text mining [Mühlroth and Grottke, 2018] while articles on the subject of emerging trend detection account for three times as much. In total, this review found 86 articles on these subjects in this period. Hence, we can see that this research field is still young. Nonetheless, it is a growing field; from our own bibliometric analysis out of all the articles published about weak signals since 2010, 75% of them have been published in the period 2016-2022.

One limitation of the triadic model is that it is not clear what weak signals are and how each dimension can be quantified. To address this, Yoon [Yoon, 2012] proposed a system for detecting weak signals by quantifying the triadic model, modeling weak signals as manually selected keywords and focusing on the objective dimensions: the signal and the issue. Term frequency was used for measuring the signal and document frequency for the issue. They analyzed these measures through time and came up with two criteria to find weak signals. One, weak signals have low frequencies (term and document). And two, these frequencies have recently been increasing. However, a manual selection of keywords gives up all hopes of finding unknown signals and single words are often not flexible and expressive enough. Moreover, we argue that the use of document frequency is highly correlated to term frequency and does model the issue. E.g. a document may discuss multiple issues and multiple documents may discuss the same issue.

Park and Cho [Park and Cho, 2017] applied Yoon methodology on smart grid technology. While Lee and Park 2018 [Lee and Park, 2018] applied Yoon's methodology on ethical problems for AI. Both improved upon Yoon's ideas by developing a method for keyword extraction that relies less on manual work. More importantly, [Lee and Park, 2018] used a clustering technique to extract sets of keywords as topics based on co-occurrence rather than focusing on single keywords. Kim [Kim et al., 2016] expanded once more on Yoon's ideas, and created three keyword maps: a cluster map, an intensity map, and a relationship map. The cluster maps show topics, the intensity map finds weak signals, and the relationship map finds similarities between keywords. Their results have shown to be more interesting with topics such as "killer robots" and "legal liabilities" being classified as AI weak signals.

As we observe, the triadic model has been relatively well-adopted by the literature on weak signal detection [Mühlroth and Grottke, 2018]. However, many studies focus on keywords rather than topics, which are more expressive and flexible. Furthermore, most of these studies have not applied the model in a novel way and are mainly a continuation of Yoon's idea. These studies show interesting results even though none provided a quantitative evaluation.

Nonetheless, some studies have taken a different approach that does not rely on the triadic model. One paper proposed the novelty model, in which he argued that weak signals have two main characteristics: rarity and paradigm unrelatedness [Kim and Lee, 2017]. To measure these characteristics, he used the local outlier factor (LOF) method to identify local outliers based on the frequency of keywords in patent and futuristic news articles. The LOF values were then used to plot the rarity and paradigm unrelatedness of both documents and keywords in two 2D maps. Quadrants were arbitrarily created in the maps to categorize weak, strong, feeling, and stagnant signals based on their rarity and paradigm unrelatedness.

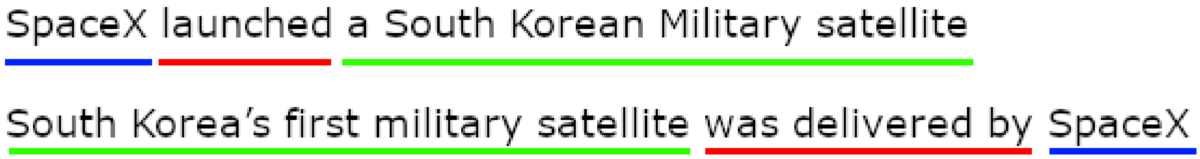
Another paper used patent citation analysis and clustering techniques to identify weak signals in the form of emerging technologies [Breitzman and Thomas, 2015]. His approach involved identifying "hot patents," or highly cited patents with a high proportion of recent citations, and "next-gen patents," or recent patents that cite currently hot patents. He argued that hot patents can serve as catalysts for emerging technologies, which manifest as next-gen patents clustering around hot patents during their "heating period." Two features of next-gen patents were used to rank them efficiently as potential weak signals: the proportion of citations from government-funded patents and the proportion of citations from research papers. The results of this approach showed that next-gen patents with high scores were 50% more likely to be cited in the next five years than the average patent, and were twice as likely to be cited as non-next-gen patents in the same time frame.

From the above, we can observe that temporality is an important component that is missing the triadic model of Hiltunen. Indeed, weak signals suggest the emergence of new trends. Therefore, it is interesting to not only understand weak signals but also to monitor their evolution. Hence, through our conceptual understanding of weak signals in this Chapter, we have set the stage for the overall methodology of this thesis that will be described in Chapter III.

2 Event & Entity Coreference Resolution

2.1 Introduction

Coreference Resolution (CR), a fundamental task in NLP, aims to identify sequences of text that refer to the same real-world object or occurrence. This task is often sub-divided into Event Coreference Resolution (EvCR) and Entity Coreference Resolution (EnCR) which involve clustering event and entity mentions respectively [Barhom et al., 2019, Lee et al., 2017, Joshi et al., 2019, Choubey and Huang, 2017, Kenyon-Dean et al., 2018]. Figure 2.1 illustrates this concept of coreference with event and entities mentions with identical meanings in two distinct sentences.



SpaceX launched a South Korean Military satellite

South Korea's first military satellite was delivered by SpaceX

Figure 2.1: Two coreferent event mentions with colors indicating associated coreferent entity mentions.

Event mentions represent textual descriptions of real-life events, typically consisting of a trigger word (often a verb) and a set of arguments (entities). For instance, in Figure 2.1, the trigger word "launched" is accompanied by arguments "SpaceX" and "a South Korean Military satellite". Four argument types can be identified: Arg0, Arg1, location, and time, as defined in [Barhom et al., 2019]. Arg0 (resp. Arg1) refers to the nearest entity to the left (resp. right) of the trigger word.

Notably, the task of Coreference Resolution can also be divided into Cross-Document Coreference Resolution (CD) vs Within Document Coreference Resolution (WD). While the WD setup aims to cluster mentions inside a single document, CD aims to cluster mentions across them.

In this Chapter, we will explore models capable of addressing the tasks of EvCR and EnCR, followed by a discussion of models capable of addressing both. In our methodology, we initially aimed to use CR to accurately measure the issue by detecting duplicate event that would have been extracted from a corpus. We will explain in Chapter III and 5 why this was not reasonably feasible and how this affected our final methodology.

2.2 Event Coreference Resolution

In one study [Peng et al., 2016], the authors have put forth an unsupervised approach to Event Coreference Resolution. They used the cosine similarity on the embeddings of mention pairs for clustering. The authors conducted experiments with a variety of embedding systems, such as ESA, Brown cluster, Word2vec, and DEP, to evaluate their methodology. Although their system does not set new standards, it achieves comparable performance with existing systems at the time.

In another study [Choubey and Huang, 2017], the authors highlight that while most studies focus on WD EvCR, CD EvCR is inherently more challenging. They propose an iterative solution that alternates between CD and WD resolutions as a clustering problem using pairwise classifiers. This approach, which emphasizes the importance of both local and global context, operates in two stages: first, CD/WD merges are suggested by a pairwise classifier, and in the second stage, second-order relations across clusters are utilized for further merging. The iterative alternation between CD & WD merges continues until no further merging is possible.

Finally, another study proposes a novel approach to CD/WD EvCR [Kenyon-Dean et al., 2018]. They introduce an alternative to the traditional pairwise clustering, creating embeddings that draw coreferent elements closer together through the use of cluster-oriented regularization during training. This allows for traditional hierarchical clustering methods to be applied.

2.3 Entity Coreference Resolution

In one study [Wiseman et al., 2016], the authors identify the limitations of traditional EnCR systems, which are mostly local. They propose a model that learns representations of mentioned clusters by sequentially embedding them using a recurrent neural network. Moreover, they argue that while EnCR is fundamentally a clustering task, it can be approached by identifying antecedents and recovering clusters from chains of antecedents.

In another paper [Lee et al., 2017], the authors propose an end-to-end neural coreference resolution model that eliminates the need for a syntactic parser. The model jointly learns to find and cluster mentions, operating at the span level. Using embeddings and an attention mechanism, the model determines whether each span represents an antecedent. Due to computational constraints, the model applies a scoring system to classify spans as entities and prunes those with low scores. The model also restricts the number of considered antecedents for each span. Despite aggressive pruning, the model maintains a high recall rate.

2.4 Joint Models

In one study [Lee et al., 2012], the authors address the limitation of traditional coreference resolution systems that primarily focus on EnCR. They present a holistic approach that considers both events and entities. Their approach utilizes an iterative algorithm that constructs clusters of entity and event mentions using linear regression to model cluster merge operations. This algorithm also allows for the interplay of information between entity and event clusters via features that model context using semantic role dependencies.

In a subsequent study [Barhom et al., 2019], the authors aim to address the challenge of CD/CR, a task that is vital but often overlooked in favor of within-document coreference. Recognizing that both entity and event coreference are crucial for CD/CR, the authors propose a novel joint neural architecture where the representation of an event (entity) mention considers other entities (events) connected to it via predicate-argument structure. This model cluster mentions based on a learned pairwise mention coreference scorer.

2.5 Evaluation Metrics

Evaluating the performance of coreference resolution systems is essential to measure their effectiveness and guide their development. However, assessing coreference resolution is not straightforward due to the complexity of the task setup. Indeed, in a simplified understanding of the task, given a set of mentions M, there is B_M possible clustering configurations of these mentions corresponding to the Bell number of M. There is only one correct configuration (T) where each cluster contain only and all mentions that are coreferent. However, compared to a simpler classification or regression setup, there is no clear definition of distance between T and any other $B_M - 1$ possible configurations.

Moreover, coreference resolution models do not directly produce clusters of mentions. Instead, they tend to resolve links between pairs of mentions thus producing a graph of mentions (nodes) and links (edges). Afterwards, we can recover clusters of mentions by extracting the connected components of the resulting graph. Here, we do not have B_M possible configurations of links but rather $2^{((M-1)*M)/2}$ configurations given than fully connected graph as $(M-1*M)/2$ edges. In this setup, the problem worsen since one wrong link can connect (resp. split) two separated (resp. connected) components of the graph. Depending on the size of these components, one link may have a small impact or a large impact on performance. Additionally, adding one wrong link between two mentions which are already inside a connected component does not change the final clusters and thus has no more impact on the final set of cluster. Hence, one link can impact performance in an arbitrary manner. Therefore, defining a metric for evaluating the performance of coreference resolution is a complex combinatorial problem with no trivial solution.

Hence, many different evaluation metrics, such as MUC, B3, CEAF and BLANC, have been proposed to address coreference resolution evaluation. However, these metrics are complex solution to a combinatorial problem and thus their definition is outside the scope of this thesis. Moreover, past research demonstrated that these metrics are neither reliable nor easily interpretable [Moosavi and Strube, 2016]. As a consequence, researchers have been using the CoNLL measure defined as the average of MUC, B3 and CEAF as an ad hoc solution to the unreliability of these metrics. A new metric, LEA, has been proposed to tackle the problem of reliability [Moosavi and Strube, 2016]. However, it remains difficult to interpret and has not been accepted by the latest state of the art in coreference resolution [Held et al., 2021].

As we can see, Coreference Resolution is a complex and deep field of study. Nonetheless, our inability to accurately measure the performance of a model makes it challenging to rely on them. Moreover, the supervised nature of the task makes it difficult to scale to new data while ensuring generalizability and cost effectiveness. Furthermore, there are other problems of consistency in terms of the definition of the task as we will discover in Chapter 5. Consequently, our final methodology for weak signal detection does not make use of CR.

3 Word Embeddings

3.1 Introduction

To perform CR and many other tasks in NLP such as topic tracking [Liu et al., 2020a, Xu et al., 2019, Zhu et al., 2016, AlSumait et al., 2008, Allan et al., 2003, Zhe et al., 2011], we rely on Word embeddings. They are a set of methods that allows us to translate natural language into a vector space. Precisely, they provide a way to encode words or sequences of words into dense numerical vectors called word embeddings [Mikolov et al., 2013]. These embeddings encode words in such a way that the semantic relationships between words is translated into algebraic relationships. A trivial example is that v(King)-v(Man)+v(Woman) = v(Queen) where v(*) is the function that maps each word to a word embedding.

Words embeddings have revolutionized the field of NLP as they allow for a dense representation of words that can be fed directly into machine learning models such as deep neural networks. Hence, since their inception, their use only increased and many types of word embeddings have been proposed. Specifically, the literature generally distinguishes between three families: *static*, *contextual*, and *character* embeddings [Almeida and Xexéo, 2019, Liu et al., 2020b, Dos Santos and Zadrozny, 2014]. Nonetheless, there exist many other kinds of word embedding that are more specific to some applications such as phonetic-based [Liu et al., 2018] and non-euclidean embeddings [Nickel and Kiela, 2017].

We will use word embeddings in Chapter 5 to perform Event & Entity Coreference Resolution and in Chapter 7 for Topic Tracking.

3.2 Static Embedding

Static embeddings were the first kind of word embedding proposed [Mikolov et al., 2013, Bojanowski et al., 2017, Pennington et al., 2014]. To transform words into dense numerical vectors, they start with a sparse representation of these words called one-hot encoding. Given a vocabulary of size V. The one-hot encoding vector for a word i have V dimensions and is full of zeros except at index i which is set to one. Some embeddings start with the one-hot encoding of characters or part of words [Bojanowski et al., 2017, Dos Santos and Zadrozny, 2014]. These one-hot vectors are sparse, extremely large, and by definition linearly independent. Through, word embedding methods, we will see how we can transform these vectors into smaller and denser vectors whose linear relationships encode semantic relationships between the words they represent. The size of these embedding vectors tends to be 300 as larger static embeddings tend to provide only marginal gains in performance [Patel and Bhattacharyya, 2017].

The fundamental assumption behind word embeddings is called the distributional hypothesis [Mikolov et al., 2013] which states that the meaning of a word can be inferred from the context in which it appears, and that words that appear in similar contexts tend to have similar meanings. For example, the embedding for "car" and "truck" might be more similar than "car" and "star" because "car" and "truck" often appear in text around similar words such as "roads" and "traffic".

Word2Vec was one of the first word embedding proposed [Mikolov et al., 2013]. Word2Vec produces embeddings through a neural network architecture that is similar to an auto-encoder [Tschannen et al., 2018]. Specifically, there are two versions of Word2Vec : CBOW and Skip-gram. CBOW models are trained by trying to predict a word given the words that surrounds it while Skip-gram models are the opposite; they are trained by trying to predict the words that surround a given word. Since the architecture of Word2Vec is similar to an auto-encoder, there is many fewer neurons in the middle of the network than on the input and output side. This forces the network to compress the information into a compact representation. After the training process, it is this hidden representation that is used as a word embedding.

GloVe [Pennington et al., 2014] was proposed soon after Word2Vec as another word embedding model. It uses a co-occurrence matrix to learn word representations. The model constructs a co-occurrence matrix from a large corpus, where each cell in the matrix represents the number of times a word appears in the context of another word. GloVe then factorizes this matrix using matrix factorization techniques to obtain word embeddings that capture both the local and global context of words. Unlike CBOW and skip-gram models, GloVe takes into account the global statistics of the corpus, such as the frequency of word pairs, to learn word embeddings.

FastText [Bojanowski et al., 2017] is a more recent static embedding that works similarly to Word2Vec. Its main difference is that it learns the embedding of sub part of words. For example, the word "fabulous" could be cut into the pieces <fa,fab,abu,bul,ulo,lou,lous,us>. Through this method, FastText is capable of better-capturing information about word morphology which is useful in inflected languages such as English. For example, the words "eat" and "eating" have a similar but slightly different meanings through the suffix "-ing" which FastText is supposed to better capture. Moreover, while Word2Vec is not capable of providing an embedding for a word that is outside its training set, by using sub-parts of words, FastText is capable of providing embedding to these out-of-vocabulary words.

Finally, character embeddings are a sub-class of static embeddings [Dos Santos and Zadrozny, 2014, dos Santos and Guimarães, 2015]. Instead of using words or sub-part of words, they encode each individual character of a word into a one-hot vector. Hence, for a given word we have a set of one-hot encoded vector that forms a matrix. A CNN with a 1-d kernel is then used to find a lower representation of these words by training it in conjunction with a task such as text classification. While these embeddings are extremely useful for encoding out-of-vocabulary words, many words have totally different meanings with only a one character difference such as "dock" and "duck". The size of these embedding vectors tends to be 50 because they seem to contain less information [Patel and Bhattacharyya, 2017].

3.3 Contextual Embedding

While static embeddings provide a good representation of words, they are not capable of handling word polysemy; the existence of multiple meanings for a given word. For example, in the sentences "I deposited my paycheck at the bank" and "The river bank is eroding", the word "bank" has two different meanings. A static word embedding would still only provide a unique representation for "bank". In contrast, contextual embedding provides a vector representation that is relative to the context which means the word "bank" will have different representations in these two sentences. To do so, contextual embeddings use static embeddings as input. The size of these embedding vectors tends to be larger than static embeddings (i.e. 1024) because they provide more information.

ELMo was the first contextual embedding that was proposed [Peters et al., 2018]. The architecture is a Bi-LSTM trained based on a bi-directional language modeling task. This means each word is predicted given all others. Hence, for a given sentence, at the output of the Bi-LSTM, the embedding of each word is affected by the embedding of the words around it. This also means that contextual embeddings must be applied to sequences of words to be effective. Moreover, due to the nature of the Bi-LSTM, words that are further apart affect each other less.

BERT followed up soon after and proposed a transformer architecture to train word embedding based on a masked language modeling task. This means that for each input sequence, a few words are masked and must be predicted. The appeal of transformers is that they use the attention mechanism to inject contextual information into contextual embedding. This means that, contrary to LSTM solutions, the distance between words is not a factor in deciding what matters in the context.

Finally, GPT provides a similar method for producing contextual word embeddings. However, it is trained using an auto-regressive language modeling task. This means GPT is trained to predict the next word for a given sequence making GPT well suited for text generation tasks.

3.4 Studies on Embeddings' Performance

One study [Gromann and Declerck, 2018] demonstrated that FastText surpasses Polyglot and Word2Vec in ontology alignment, achieving an F1 score of 0.812 compared to 0.675 and 0.750 respectively. The study, which utilized the Global Industry Classification Standard and the Industry Classification Benchmark ontologies, also highlighted FastText's superior capability to manage out-of-vocabulary words.

In a distinct study [Berardi et al., 2015], the authors found that Word2Vec outclassed Polyglot and GloVe in a word analogy test, obtaining an accuracy of 43.63% in comparison to 40% and 30.21% respectively. This experiment employed Wikipedia and a collection of Italian literature, predominantly novels, as data sources.

Other researchers [Li et al., 2018] discovered that GloVe exceeded both FastText and Word2Vec in a tweet classification task, especially when trained on specific corpora, specifically CrisisLexT6, CrisisLexT26, and 2CTweets. These three studies demonstrate that there is no general rule as to whether one static embedding is better than the others.

We can also review the performance of word embeddings on our task of interest : Event & Entity Coreference Resolution (EvCR and EnCR). Regarding EnCR, one study [Lee et al., 2017] leveraged GloVe for word representation in conjunction with a Bi-LSTM and attention mechanisms. Their model achieved state-of-the-art performance with a 68.8 F1 score on the CoNLL-2012 corpus. While another study [Joshi et al., 2019] reported improved EnCR results when employing BERT over ELMo, with increases of +3.9 F1 in OntoNotes and +11.5 F1 in GAP.

In the EvCR domain, one study [Choubey and Huang, 2017] utilized GloVe for EvCR on the ECB+ corpus [Cybulska and Vossen, 2014]. They developed a joint modelling approach for within and cross-document EvCR, achieving state-of-the-art performance. A subsequent study [Barhom et al., 2019] also used this corpus, proposing an EvCR/EnCR model based on ELMo [Peters et al., 2018], GloVe [Pennington et al., 2014], and a fine-tuned character embedding. Their model, which jointly performs EnCR and EvCR, achieved an impressive 79.5 F1 score in EvCR.

As we can observe, we cannot point to a single embedding as the overall best performing solution to all tasks. Specifically, there is no study comparing the various embeddings in Coreference Resolution. We will address this knowledge gap in Chapter 5. Morover, we will see how these embeddings can be used to track topics over time in Chapter 7.

4 Topic Models

4.1 Introduction

The core of this thesis relies on topic models which are a broad class of unsupervised machine-learning techniques aimed at discovering hidden semantic structures in a corpus of documents. These models transform unstructured text data into structured topics, which are distributions over words that capture the thematic content of the documents [Blei et al., 2003].

The essence of topic modeling is to represent documents as mixtures of latent topics. A topic, in this context, is a collection of words that are semantically related. For example, words like "election", "vote", "democracy", might constitute a "politics" topic, while "neutron", "quantum", "particle" might constitute a "physics" topic. The underlying assumption is that a document is a mixture of various topics, and each word in the document is attributable to one of the topics in that document.

Topic models have become a critical part of the toolkit for researchers dealing with large text corpora, enabling tasks such as document summarization [Yang et al., 2015], environment scanning [Gregoriades et al., 2021, Kim et al., 2020], understanding employee and customer satisfaction [Korfiatis et al., 2019, Bastani et al., 2019] among others [Hong et al., 2011, Zhou and Chen, 2013]. Since the introduction of the first probabilistic topic model, Latent Dirichlet Allocation (LDA) [Blei et al., 2003], numerous variants have been proposed to address specific challenges. In this Chapter, we will review these different sub-types namely, flat topic models, hierarchical topic models, temporal topic models, and correlated topic models. Additionally, this Chapter will review other related sub-fields such as topic tracking which aims at tracking the evolution of topics over time, and the topic evaluation methods. The methods and tools presented in this Chapter will form the foundation of this thesis.

4.2 Flat Topic Models

Latent Dirichlet Allocation (LDA) is considered the first topic model and remains popular to this day due to its simplicity [Blei et al., 2003]. LDA is a generative probabilistic model which assumes that each document in a corpus is a mixture of a finite number of topics. Each topic is represented as a distribution over a vocabulary, and each document is represented as a distribution over topics. LDA employs a Dirichlet prior to modeling these distributions. The goal of LDA is to infer the latent topics and the document-topic distributions given the observed documents, which is typically achieved through stochastic variational inference or Gibbs sampling. The main weakness of LDA is that it requires the user to specify a predefined number of topics to be extracted. However, such information is usually not known in advance. Consequently, LDA requires a long model validation step to determine the number of topics.

As a solution, the Hierarchical Dirichlet Process (HDP) model was proposed as an extension of LDA [Teh et al., 2006]. Unlike LDA, which requires a pre-specified number of topics, HDP automatically determines the appropriate number of topics during training. HDP achieves this by replacing the Dirichlet prior to LDA with Dirichlet Processes. Otherwise, HDP operates similarly to LDA.

The Correlated Topic Model (CTM) is another extension of LDA that assumes topics are not independently distributed in documents [Blei and Lafferty, 2007]. This is a more realistic assumption in many scenarios, such as in a document about genetics being more likely to also discuss diseases than astronomy. The CTM achieves this by using the logistic normal distribution to model variability among topic proportions, as opposed to the Dirichlet distribution used in LDA.

4.3 Hierarchical Topic Models

Methods like LDA and HDP, while effective in many contexts, primarily generate a flat topic structure. Hence, they treat all topics as being at the same level of abstraction. However, in complex or large-scale text corpora, topics often exhibit inherent hierarchical relationships. For example, in a corpus about science, "biology" and "physics" could be top-level topics, with "genetics" and "quantum mechanics" as respective subtopics. Flat models like LDA and HDP cannot capture such hierarchical associations. Recognizing this limitation, the focus has shifted toward the development of models that can learn hierarchical topic structures. By extracting topics and subtopics, these hierarchical models enable a more detailed and nuanced understanding of the thematic structure of a corpus. For our purposes, hierarchical models provide a more fine-grained analysis which could help uncover weak signals and micro trends.

The state-of-the-art for hierarchical topic modeling is nHDP [Paisley et al., 2015]. It models topic hierarchy by defining a potentially infinite tree where each node corresponds to a topic. At each branch of the tree, we exactly have the HDP model. The difference is that, when a word is assigned to a topic during training, there is a chance to go deeper in the tree based on a Bernoulli distribution. If we do go deeper, we repeat the HDP algorithm with a sub-corpus made up of the documents and tokens assigned to the selected topic.

The Nested Hierarchical Dirichlet Process (nHDP) represents the state-of-the-art model for hierarchical topic modeling [Paisley et al., 2015]. In the nHDP model, topics are arranged in a potentially infinite tree structure, where each node signifies a topic. Each level of the tree resembles the HDP model. What differentiates nHDP is that for each word, the model decides iteratively if that word needs to be assigned to a topic deeper in the tree based on a Bernoulli distribution. If the decision is to delve deeper, the HDP algorithm is recursively applied to a sub-corpus comprising the documents and tokens associated with the topic where the decision is made. Once again, the word is then assigned to a sub-topic based on the HDP model. This mechanism allows nHDP to grow automatically a hierarchically organized set of topics, from broad themes to more specific sub-themes, thereby providing a granular view of the corpus's thematic landscape.

Previously, the nested Chinese Restaurant Process (nCRP) [Blei et al., 2004] was proposed. It is the predecessor of nHDP and works similarly. Nevertheless, it does not model the document-topic distribution as in nHDP. This means that a document-topic distribution is limited to a singular path from the root of the tree to a branch.

Other topic models have been proposed to model hierarchy. For example, the Large-Scale Hierarchical Topic Model (LSHTM) [Pujara and Skomoroch, 2012] recursively applies LDA to the sub-corpus defined by the topics of the previous LDA application. However, it means that it inherits from the limitations of LDA that the number of topics to extract needs to be predefined.

Finally, the hierarchical Pachinko Allocation Model (hPAM) [Mimno et al., 2007] capture arbitrary topic relations by modeling topics as a Directed Acyclic Graph (DAG) instead of a tree structure. This structure allows topics at lower levels to share multiple parent topics. While this provides less trivial relationships between topics, it is harder to display and navigate due to the complex web of relationships.

4.4 Temporal Topic Models

While flat and hierarchical topic modeling approaches have provided valuable insights into the thematic structure of text corpora, they do not model the temporal dimension. Hence, In parallel to research on hierarchical topic models, other researchers have begun to explore the temporality of topics, aiming to understand not only what topics are discussed, but also when they occur and how they evolve over time. The integration of this temporal aspect is of particular importance when analyzing trends, as it enables us to monitor topics, identify emerging topics, and detect events that are represented by topics localized in time.

The Topic over Time (ToT) model extends LDA by incorporating temporal information into the topic discovery process [Wang and McCallum, 2006]. In ToT, each document is accompanied by a timestamp, which allows us to fit a beta distribution for each topic. This distribution represents the temporal aspect of the topic and is jointly optimized with the topic-word distributions during the learning process. Consequently, the ToT model is capable of revealing topics that are localized in time (i.e. events) or topics that demonstrate clear trends of growth or decline over time. This ability to capture both events and trends adds an extra layer of richness to the understanding of topics in a corpus.

The Multiscale Topic Tomography (MTT) model is another approach to temporal topic modeling that offers an understanding of topics across multiple time scales [Nallapati et al., 2007]. In MTT, a separate tree is constructed for each topic, where nodes at varying depths represent the topic's behavior at different time granularities. Specifically, deeper nodes in the tree correspond to finer temporal resolutions while horizontally, at each level nodes represent different time periods.

The Dynamic Topic Model (DTM) captures the evolution of topics in a corpus over time [Blei and Lafferty, 2006]. In DTM, the corpus is divided into discrete time slices, with the topic distributions in each slice being influenced by those in the preceding slice. The initial time slice is processed using standard LDA, while subsequent slices incorporate information from the previous slice, using it as a prior in the Bayesian inference process. This sequential approach enables DTM to model the dynamic nature of topics, tracking how they change and evolve across different periods.

4.4.1 Topic Tracking

Topic tracking is a key aspect of temporal topic modeling, involving the monitoring of topic evolution over a given timeframe. Different models offer varied degrees of temporal analysis. For instance, models such as MTT and DTM provide detailed insights into how topics evolve over time, considering changes across different time scales. On the other hand, the ToT model primarily offers a temporal distribution, pinpointing when topics occur but providing less information on their progression. Notably, ToT uniquely excels at identifying temporally localized events. Nonetheless, tracking the evolution of topics can also be conducted as a post-training analysis, offering additional flexibility in understanding topic dynamics over time.

Numerous researchers have explored the idea of linking topics extracted independently from different time periods to track their evolution over time [Zhu et al., 2016, Xu et al., 2019, Liu et al., 2020a]. In one study, the authors applied Latent Dirichlet Allocation (LDA) to each year's data independently and then linked the resulting topics using the Jensen-Shannon Divergence (JS), a measure of similarity between probability distributions [Dagan et al., 1997, Zhu et al., 2016]. Another study used a similar method but added an extra layer of complexity by clustering linked topics [Xu et al., 2019]. This means that once two topics were linked, they formed a cluster, and subsequent topics were compared to the entire cluster rather than just the preceding topic. Yet another study proposed a tracking method using the JS divergence but did not constrain linkage to a one-to-one mapping, thereby allowing for the fusion and splitting of topics [Liu et al., 2020a].

In the context of this thesis, we have opted for a posteriori topic tracking, primarily due to our objective of combining temporal and hierarchical information into a single topic model, as discussed in Chapter 6. Combining a hierarchical structure, as in the nHDP model, with a temporal structure, akin to the MTT model, could potentially overcomplicate the model. Additionally, we sought to create a model that could be scaled up effectively. This necessitated avoiding models like DTM, which are inherently sequential and cannot be parallelized. Consequently, we converged on integrating the ToT and nHDP models, a combination that enables us to train various time slices in parallel and conduct a posteriori topic tracking, which can also be performed in a parallel manner. This approach offers a balance between complexity and scalability while still capturing both temporal and hierarchical dimensions of topics.

4.5 Topic Model Evaluation

Evaluating the performance of topic models is a critical aspect of topic modeling research, as it facilitates the comparison of different models and the assessment of their effectiveness. For a long time, perplexity was the standard metric for such evaluations. It gauges the likelihood of the training data having been generated by the trained topic model. However, it was recently found that perplexity does not necessarily correlate well with human judgment [Chang et al., 2009]. This revelation spurred the development of alternative evaluation techniques, although no single method has emerged as a new standard.

One of these alternative methods is Topic Coherence [Newman et al., 2010], which involves calculating similarity scores between the top N words of a topic. The computation is done as follows: $\sum_{i<j} score(w_i, w_j)$, where w_i is more frequent than w_j. This method is flexible, accommodating a variety of scoring functions. UCI and UMass, both utilizing word co-occurrence for scoring, are among the most popular. UCI is an extrinsic measure based on an external dataset such as Wikipedia articles, while UMass is intrinsic and uses the training corpus. Other scoring functions, such as the cosine similarity of word embeddings, can also be employed. The average coherence score of the topics gives the topic coherence score of a model. However, recent studies question the correlation between coherence measures and human ratings [Newman et al., 2010, Doogan and Buntine, 2021, Bhatia et al., 2017].

The Word Intrusion task is another recently developed evaluation method. This technique involves embedding an intruder word within the top word list of a topic and having individuals identify it [Chang et al., 2009]. This intruder word is randomly selected from a pool of words that have a low probability in the current topic but a high probability in another, to avoid the inclusion of rare words. The premise is that high-quality topics will allow annotators to easily spot this intruder. The final score of this evaluation method corresponds to the average classification accuracy determined by human annotators. A subsequent study demonstrated that this Word Intrusion task can be automated [Lau et al., 2014]. However, we will show in Chapter 8 that the Word Intrusion task may not be reliable either for hierarchical topic models.

A more direct understanding of a model's performance can be obtained through qualitative analysis of the extracted topics, a method adopted widely. However, this approach, compared to quantitative measures like coherence and perplexity, is susceptible to cherry-picking, especially when a large number of topics are extracted.

Hence, despite the variety of evaluation methods available, none have proven to be consistently reliable. Moreover, this problem worsen for Hierarchical Topic Models (see Chapter 8). Furthermore, none of these methods adequately assess comprehensively the quality of the results: Do we extract every topic we expect to extract? How effectively do we extract them? Do we extract unanticipated topics? Is the hierarchy produced coherent? Thus, it is evident that new tools for evaluating topic models, particularly hierarchical ones, are required. Chapter 9 will adress these challenges and propose a novel approach to Topic Model evaluation.

Part III

METHODOLOGY

In this Section, we will outline the methodology used throughout the thesis, which aims to develop a method for weak signal extraction based on the theoretical elements discussed in the literature review on weak signals. Our goal was to create a scalable and unsupervised approach that could extract and track emerging topics over time, identify topic interactions, and provide a platform for scenario planning and decision-making. Through a series of iterations, we developed a novel methodology that combines a hierarchical and temporal topic model (HTMOT) with topic tracking and correlations. Additionally, we devoted a portion of the thesis to developing a methodology for evaluating the performance of the topic model, which led to the creation of a new approach based on supervised evaluation for unsupervised models.

In the initial phase of the methodology, our aim was to quantify the dimensions described by the Triadic model of Future signs [Hiltunen, 2008]. To achieve this, we attempted to measure the dimensions using the following method:

1. To extract potential future signs, we used a topic model. We found that using topics provides a flexible approach compared to traditional keyword searches. Additionally, while keywords must be selected, topics can be discovered, which is advantageous when trying to uncover the unknown.

2. We measured the *signal* through the frequency of a topic describing a sign. This is similar to the term frequency measure proposed by Yoon [Yoon, 2012], but in the context of topics.

3. For measuring the *issue*, we aimed to extract events described in the data. This involved computing the number of events in a topic, which required the removal of duplicates through the event coreference resolution task.

4. We chose sentiment analysis as the method for measuring *interpretation*, as it is the obvious choice in this context.

Based on this approach, we began investigating possible methods to measure the *issue*, which ultimately led to our first paper on the subject of Event Coreference Resolution (see Chapter 5).

Despite the successful publication resulting from this approach, we encountered some drawbacks when attempting to quantify the issue dimension. First, this required us to extract events from the data and then remove duplicates through coreference resolution. However, even with state-of-the-art methods, the high pipeline error resulting from the extraction and subsequent coreference of events would have led to poor performance. Second, the lack of a consistent definition for these tasks in the literature made it difficult to find relevant datasets and benchmarks for our specific case. Finally, the thesis aims to develop a method that can operate at a large scale on any type of data, and using supervised solutions for event extraction and coreference resolution may lead to poor generalization on unseen data, especially given the lack of large datasets available.

In the subsequent years, we shifted our focus towards unsupervised methodologies, aimed at tackling the challenge posed by the limited availability, high cost and generalizability issues associated with labeled data. This pivot led us to reevaluate our focus on sentiment analysis, a domain that had already reached a high level of maturity, making it daunting to offer novel and substantial insights. Consequently, our attention turned to the refinement and innovation of evaluation techniques for topic models as well as the computation of topic correlations.

Topic correlations is also an important contribution to the overall methodology as it provides a deeper understanding of the interactions between topics. As stated in the review of the literature on weak signals in Chapter 1, the auto-catalytic nature of weak signals means that understanding their interactions is crucial to better understand their potential. Hence, we believe that computing topic correlations allows us to better understand these inter-dependencies. Nonetheless, we did not perform any significant experiments using topic correlations since we decided to focus instead on the bigger problem of topic model evaluation methods. Consequently, this subject will not be discussed further.

This final strategic decision was influenced both by invaluable input from a member of my thesis committee and by an existing opportunity in the field for impactful contributions. This opportunity was particularly evident due to the numerous limitations inherent in the existing evaluation methods, which we discussed in Section 4. Hence, in Chapter8 we provide a critic of the latest evaluation method and propose our own in Chapter9.

Therefore, while the triadic model served us well to understand weak signals and define the initial direction of research, the final methodology is not based on quantifying the three dimensions. Instead, the extraction and subsequent analysis of topics extracted from textual data became the core of the methodology. In the final version of our methodology, we developed a hierarchical and temporal topic model (HTMOT) as the core component of the system. With HTMOT, topics and sub-topics can be extracted in a hierarchical manner, and the model also incorporates a temporal component that allows it to extract specific events described in a corpus. Moreover, another module allows us to track these topics over time and detect emerging ones. With this new approach, small sub-topics (that may be localized in time) which cannot be traced back to previous periods are indicators of new emerging weak signals. In other words, the weakness of a signal (small sub-topics) and its novelty (cannot be traced back in time) are the main factors which allows us to detect weak signals. The topic correlation module in itself does not help us with the detection of weak signals but rather provide additional information about how these weak signals interact with other weak signals and larger trends. Thus, providing a platform for scenario building and strategic planning.

The overall methodology can be described as follows:

1. Extract topics using HTMOT to discover topics at a fine-grained level both temporally and thematically.

2. Use topic tracking and correlations to understand the evolution of topics and to discover emerging weak signals.

3. Use topic correlations to uncover interactions between topics, which provides a platform for scenario building.

4. Compile the extracted information on topics, their evolution, and correlations on a dashboard that allows users to navigate the results.

All the methods used in the extraction and monitoring of weak signals and emerging trends are unsupervised.

We will now describe each component of the methodology in details :

- Chapter 5 will describe Event Coreference Resolution

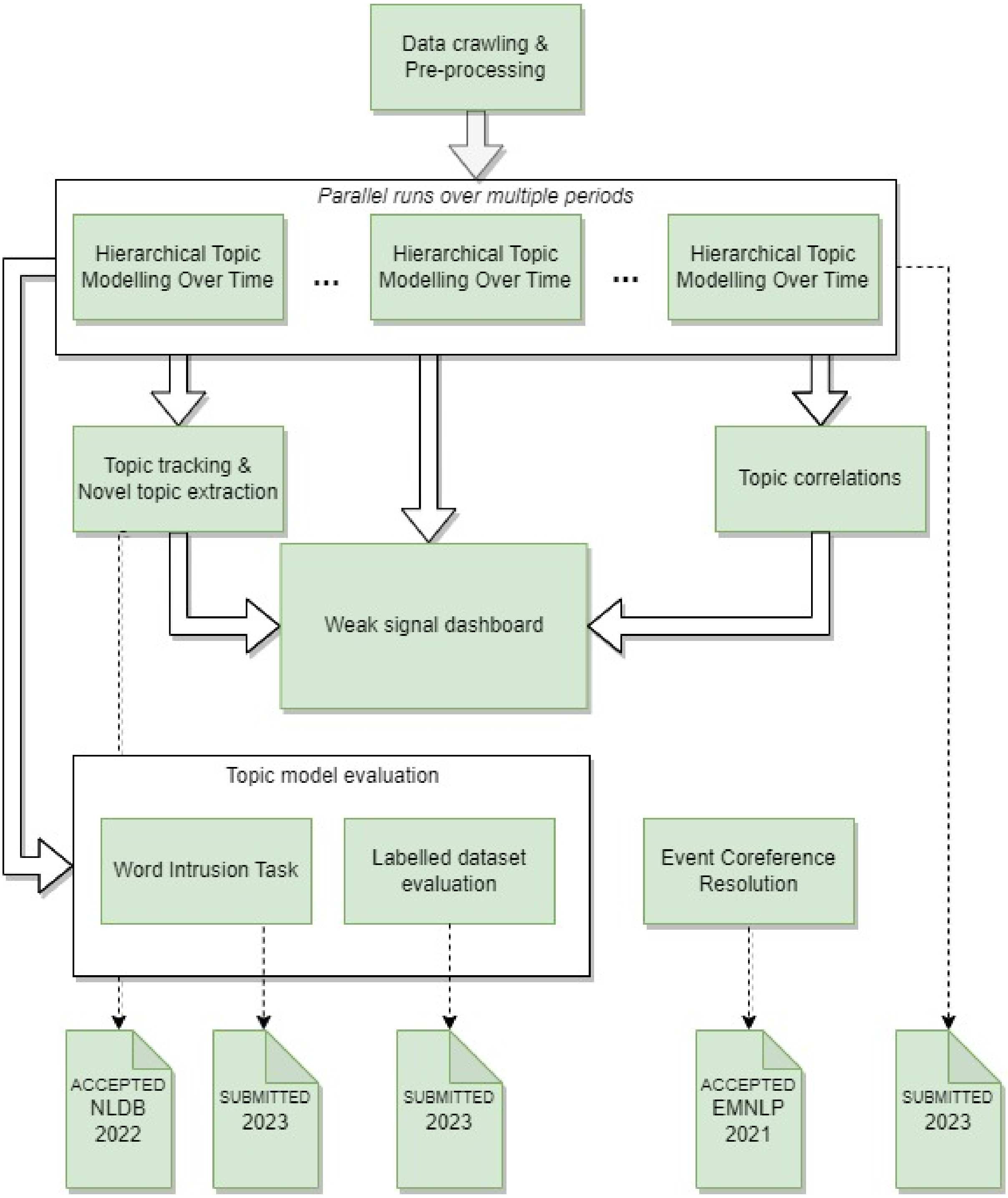

Data crawling & Pre-processing
Parallel runs over multiple periods
Hierarchical Topic Modelling Over Time
...
Hierarchical Topic Modelling Over Time
...
Hierarchical Topic Modelling Over Time
Topic tracking & Novel topic extraction
Topic correlations
Weak signal dashboard
Topic model evaluation
Word Intrusion Task
Labelled dataset evaluation
Event Coreference Resolution
ACCEPTED NLDB 2022
SUBMITTED 2023
SUBMITTED 2023
ACCEPTED EMNLP 2021
SUBMITTED 2023

Figure 4.1: Final version of the Thesis Methodology and associated papers produced

- Chapter 6 will describe HTMOT
- Chapter 7 will describe Topic Tracking

5 Word Embeddings in Event & Entity Coreference Resolution

5.1 Preamble

In this Section, we introduce our first paper, which focuses on coreference resolution. Initially, its goal was to develop an effective method for identifying and resolving references to the same real-world entities within a given text.

This paper provides a comprehensive comparison of various inputs for coreference resolution. Moreover, it shows that much smaller models can achieve competitive performance. Furthermore, it describes how larger models have been observed to train faster. Finally, it discusses the significant differences between the definition of the task of coreference resolution across various datasets and how it negatively impacts the study of the task.

5.2 Introduction

Coreference Resolution (CR) is an important NLP task. It can be subdivided into Event and Entity Coreference Resolution (EvCR and EnCR). These tasks serve as the basis for several downstream applications such as information extraction, text summarization, machine translation, and text mining [Humphreys et al., 1997, Azzam et al., 1999, Miculicich Werlen and Popescu-Belis, 2017, Su et al., 2008].

State-of-the-art methods for CR[Barhom et al., 2019, Lee et al., 2017, Joshi et al., 2019] rely on various word embeddings for word representation. These embeddings are organized into three families: *static*, *contextual*, and *character* embeddings [Almeida and Xexéo, 2019, Liu et al., 2020b, Dos Santos and Zadrozny, 2014], each differing in size. Contextual embeddings are larger (1024) compared to the other families (usually 300 for static and 50 for character). They also tend to outperform the other families in most tasks but lead to larger and heavier models [Devlin et al., 2019, Peters et al., 2018]. We are thus confronted with a trade-off of performance (predictive & run-time) vs. dimensionality. Moreover, embeddings also differ within families which also leads to differences in predictive performance.

Several studies investigated how different embeddings influence the predictive performance in different tasks [Berardi et al., 2015, Gromann and Declerck, 2018, Joshi et al., 2019, Li et al., 2018]. However, the two aforementioned issues of the performance vs. dimensionality trade-off and performance variations within and across embedding families have been overlooked to a large extent, especially in coreference resolution. Literature is still unclear about which embeddings perform best in which tasks, and whether larger, more expressive embeddings should also be preferred or whether some predictive performance can be compromised for improved run time.

Thus, we seek to address two questions in the context of CR: 1) Is there a trade-off between performance (predictive & run-time) and embedding size? 2) How do the embeddings' performance compare within and across families? The current state-of-the-art in EvCR [Barhom et al., 2019] relies on three families of embeddings for word representation, and thus provides a suitable framework for addressing our research questions. Starting from the original model of Barhom [Barhom et al., 2019], we performed various experiments and ablative studies across and within each family of embeddings, resulting in 16 different models. [1]. We compared their predictive performance, size (number of parameters) , run-time, and memory usage.

We discovered a high level of diminishing returns in terms of predictive performance per embedding. The smallest model (using solely a character embedding [Dos Santos and Zadrozny, 2014]) achieves 86% of the performance of the largest model (GloVe [Pennington et al., 2014] , ELMo [Peters et al., 2018], Character embedding) with 1.2% of its size. Hence, incorporating additional embeddings leads to diminishing returns in terms of predictive performance. In addition, we found that size and run-time are weakly correlated: larger (more complex) models can converge faster (number of epochs and total training time) than smaller ones. In terms of predictive performance, we found GloVe and FastText perform best in EvCR and EnCR respectively in their family with ELMo being the best overall. Moreover, we found that the smallest aforementioned model outperforms Word2Vec (∼+10 F1), yielding predictive performance close to the previous state-of-the-art [Kenyon-Dean et al., 2018] in EvCR (68.43 vs 69 F1). Our results can have important implications for practitioners in implementing CR and other NLP models in real-life applications.

5.3 Methodology

5.3.1 Original model

Our approach is based on the state-of-the-art model of Barhom [Barhom et al., 2019], which we refer to as the ORIGINAL [2] model. This model consists of two neural networks, which jointly resolve entities and events' coreferences. Figure 5.1 shows the input structure of both networks. The two events (resp. entity) mentions embeddings to classify are in blue and the green box represents an element-wise multiplication of these embeddings. Finally, binary features indicate whether the two encoded mentions have coreferent arguments. The constituents of each mention, i.e. trigger, Arg0, Arg1, Location, and time, are represented by a static (GloVe) and a character embedding. The trigger is also represented by contextual embedding (ELMo). Furthermore, the character embedding is fine-tuned during training while the contextual and static embeddings are not. The model outputs a score between 0 and 1 with a score $>$ 0.5 indicating that the two input mentions are coreferent. For example, given the mention "the city of Liège" and it's nickname "The ardent city", we should expect the model to output a value higher than 0.5 classifying these mentions as coreferent.

[1] The relatively large number of models and experiments is one reason why we preferred to focus on a single task

[2] MODELNAME denotes a model

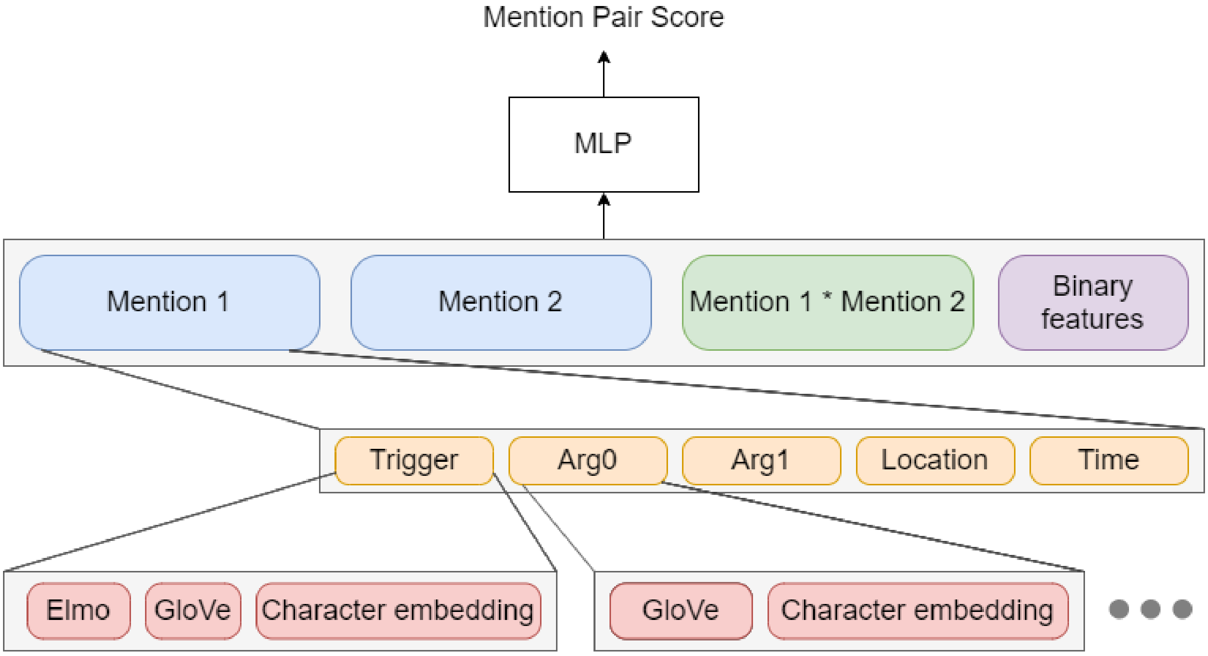


Figure 5.1: Original input structure of Barhom's model [Barhom et al., 2019].

The input dimensionality is $input = 3*(1024+5*(300+50))+200 = 8522$, where 1024, 300, and 50 are the dimensions of ELMo, GloVe, and the character embeddings, and 200 corresponds to the size of the binary features. This input is then fed into two subsequent ReLU layers with dimensions equal to half the input dimension (4261 neurons each). Since the number of parameters is proportional to the square of the input dimension, we have a model size exceeding 54 million parameters, computed as $(\frac{input^2}{2} + (\frac{input}{2})^2 + \frac{input}{2})$ [3].

5.3.2 Derived models

The gist of our methodology involves substituting and/or removing specific embeddings from Barhom's original model [Barhom et al., 2019] (which uses 3 embeddings: static=GloVe, contextual=ELMo, and character), resulting in 16 different models shown in Table 5.1. In the first group of models, one, two, or three (of the three) embeddings are removed from the original model. In the second group, the static embedding is changed to Word2Vec (Skip-gram) or FastText (other embeddings are either left unchanged or removed). Similarly, in the third group, the contextual embedding is changed to BERT or GPT-2 (other embeddings are either left unchanged or removed). Note: in Table 5.1, gray rows denote identical models.

We implemented our models using Pytorch. Models were trained and tested following Barhom's procedure [Barhom et al., 2019]. Pre-trained vectors and models were used for the embeddings. Our code is available online [4].

[3] The term $input^2/2$ corresponds to the number of connections between the input layer (i) of size $input$ and the first hidden (h_1) layer of size $input/2$. The term $(input/2)^2$ corresponds to the number of connections between h_1 and h_2 both of size $input/2$. Finally, $input/2$ corresponds to the number of connections between the second hidden layer (h_2) and the final output layer (o) of size 1.

[4] github.com/JudicaelPoumay/event_entity_coref_ecb_plus

Model	Stat.	Ctx.	Char.
Group 1: Across family study			
Original [Barhom et al., 2019]	GloVe	ELMo	✓
Contextual/Static	GloVe	ELMo	X
Contextual/Char	X	ELMo	✓
Static/Char	GloVe	X	✓
Static	GloVe	X	X
Contextual	X	ELMo	X
Char	X	X	✓
No word embed	X	X	X
Group 2: Within family study: Static			
GloVe	GloVe	ELMo	✓
Word2Vec	Word2Vec	ELMo	✓
FastText	FastText	ELMo	✓
Only GloVe	GloVe	X	X
Only FastText	Word2Vec	X	X
Only Word2Vec	FastText	X	X
Group 3: Within family study: Contextual			
ELMo	GloVe	ELMo	✓
BERT	GloVe	BERT	✓
GPT-2	GloVe	GPT-2	✓
Only ELMo	X	ELMo	X
Only BERT	X	BERT	X
Only GPT-2	X	GPT-2	X

Table 5.1: List of trained and tested model and their components. Ctx. = Contextual; Stat. = Static; Char. = Character; X/✓indicate absence/presence of an input.

5.4 Experimentation setup

5.4.1 Dataset

The dataset we use for our study is ECB+ [Cybulska and Vossen, 2014]. Together with EECB [Lee et al., 2012], it is one of the largest datasets for within and cross-document EvCR and EnCR [Lee et al., 2012, Barhom et al., 2019]. Both EECB and ECB+ are extensions of ECB [Bejan and Harabagiu, 2010] and consist of English Google News documents clustered into topics and annotated for coreference. For more details on the ECB+ statistics see table 5.2 or refer to [Barhom et al., 2019].

	Train	Validation	Test	Total
#Documents	574	196	206	976
#Sentences	1037	346	457	1840
#Event mentions	3808	1245	1780	6833
#Entity mentions	4758	1476	2055	8289
#Event cluster	1527	409	805	2741
#Entity cluster	1286	330	608	2224

Table 5.2: ECB+ corpus statistics

Other datasets for coreference resolution exist: GAP, OntoNotes, CoNLL 2012, ACE, TAC KBP and MUC. However, the definition of coreference resolution in these corpora does not suit our study and model. For example, GAP is a corpus of ambiguous pronoun-name pairs while ECB+ defines mentions cluster for events and their entities [Joshi et al., 2019]. OntoNotes annotates coreferences but does not indicate which mentions is an event and which is an entity. MUC, ACE, and TAC KBP do not provide cross-document coreferences[Lu and Ng, 2018]. Finally, while CoNLL 2012 defines an event coreference task, events represent only a small portion of all the coreferent mentions and again it does not provide cross-document coreferences [Pradhan et al., 2012]. In-depth reviews of the listed datasets are provided in [Stylianou and Vlahavas, 2021, Lu and Ng, 2018, Sukthanker et al., 2018].

5.4.2 Experiments

We performed three sets of experiments using the same hyperparameters as in the original study [Barhom et al., 2019]. Specifically, we used the default PyTorch's ADAM Optimizer with a mini-batch of 16.

The first set concerns models of Group 1 (see Table 5.1). We investigated the impact of removing one, two, or three (of the three) embeddings from the original model. Our aim was to determine the contribution of the different embeddings (static, contextual, and character) on the predictive performance of the Original model. Thus, the models will have varying sizes, translating into varying run-time and memory requirements. Therefore, for this set of experiments, we also report on model size (number of parameters), run-time (seconds), and memory usage (RAM).

The second (third) set concerns models of Group 2 (Group 3) (see Table 5.1) and aim at investigating the contributions of static (contextual) embeddings.

For the latter two experiments, we do not consider model size as all possible sizes would have been investigated in group 1. For all experiments, we will report the predictive performance achieved by the various models with the CoNLL F1 and MUC F1 metrics [Moosavi and Strube, 2016] (see Chapter 2 for more details on these metrics).

Following Barhom's original paper [Barhom et al., 2019], we can claim that a difference of 1 point between any two models is significant with a p-value < 0.001. This confirms that our results are statistically sound and not due to randomness.

5.5 Results

5.5.1 Results 1: All Embedding Families

As mentioned earlier, our aim was to investigate the contributions of the static (Glove), contextual (ELMo), and character embedding to the original model's performance via an ablative study. The predictive performance scores (CoNLL/MUC F1) of Group 1 models are in Figure 5.2, respectively from left to right.

A first observation is that the baseline performance differs between the two measures (CoNLL & MUC F1). This is due to the mentioned identification effect [Moosavi and Strube, 2016] which makes CoNLL F1 more optimistic than it should be for low-performing models. Interestingly, CoNLL seems more pessimistic than MUC for high-performing models. Moreover, Barhom's model [Barhom et al., 2019] is helped by using a gold cluster for within-document entity coreference. This explains the non-zero MUC F1 performance of the baseline on the entity coreference resolution task.

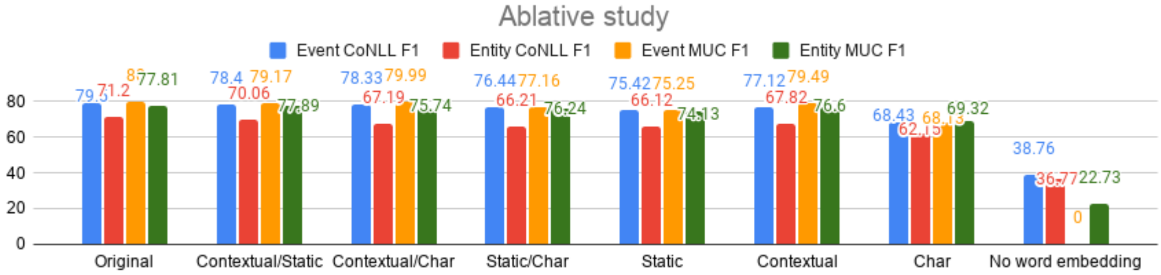


Figure 5.2: Comparing the predictive performance of the original model (using 3 embeddings) with models where we removed one, two, or all three embeddings.

Another important observation is that when using only two embeddings, the Static/Char model is the one experiencing the largest drop in performance (CoNLL & Event MUC). At the same time, when using only one embedding, the Contextual model performs best. It even outperforms the aforementioned model with *two* embeddings: Static/Char. These results lead us to conclude that contextual embeddings are the most expressive for this task. This is not surprising since contextual embeddings take context into account while static and character do not.

More interestingly, we note that removing either the static or contextual embedding results in an average performance drop of ∼2.5 and ∼4 CoNLL points respectively (see model Contextual/Char and Static/Char). However, when both are removed simultaneously, the performance drops by ∼10 CoNLL points (see model Char). That is, the sum of the losses incurred by removing either one of these embeddings (∼6.5) is smaller than the loss (∼10) incurred when both are simultaneously removed. Similarly, adding *any one* embedding to the baseline No word embedding model significantly improves the latter's performance, in the range of ∼[+27,5 to +34,7]. However, if *any one* embedding is removed from the Original model, then the latter's performance drops by a much smaller amount,∼[-1,1 to -4]. That is, removing an embedding from the Original model does not impact performance in a comparable way as adding an embedding to the baseline model. But performance does drop significantly when all embeddings are removed. In other words, we face diminishing returns in terms of performance per embedding.

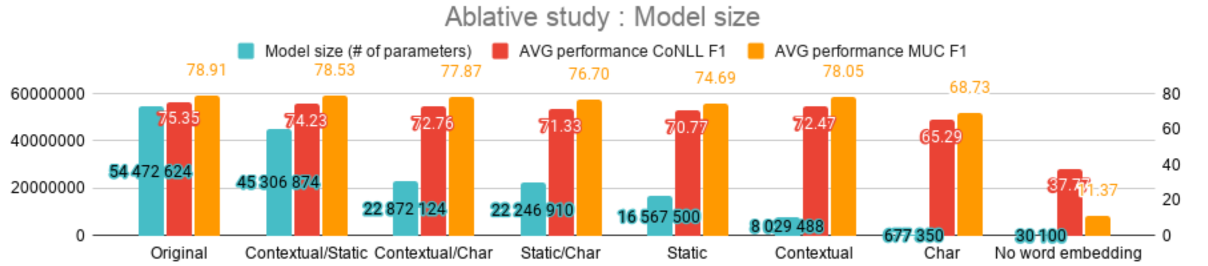


Figure 5.3: Comparing the size and predictive performance of the original model (using 3 embeddings) with models where we removed one, two, or all three embeddings. The size of each model is the number of neural connections.

Impact of Dimensionality on Model Size

As mentioned earlier, the model size is related to the square of the input, resulting in more than 54 million parameters in the Original model. Thus, an important question is whether the gains in performance of such large models outweigh the corresponding increase in size. Our observations in this respect are in Figure 5.3, depicting the model's respective size and predictive performance. We observed similar diminishing returns when considering performance relative to size, i.e. increasing the model size by incorporating larger, more complex embeddings results in modest performance gains.

The Contextual and Char models are particularly interesting. The former achieves 96% of the performance of the Original model with 14.7% of its size. While the latter, i.e. Char, achieves 86% of the performance of the Original model's performance, with only 1.2% of its size. Its performance (68.43 F1) is even comparable to that of the previous event coreference resolution state-of-the-art in EvCR (69 F1) [Kenyon-Dean et al., 2018].

Model Size & Run-Time

Our investigations on the influence of model size on run-time and memory usage revealed paradoxical results.[5] They are presented in Figure 5.4. For the run-time and memory analysis, we focus only on the largest and smallest models to have a better idea of the magnitude of differences and to avoid overcrowding the Figures.

As can be seen, the huge difference in model size (54 Million vs. 0.67 Million), does not translate into equally large differences in run-time (training & testing) - the run-time reductions afforded by the Char model are relatively modest. While the actual reasons deserve further investigation, we can posit that this could be attributed to hardware and software optimization, enabling a high level of parallelization such that larger models run comparably to smaller ones.

Paradoxically, however, the larger Original model trains in fewer epochs than the smaller Char model (14 vs. 24 respectively). In consequence, it is 21% faster to train overall (68924.8 sec. vs 87587.28 sec. or about 19h9 vs 24h19). These results confirm the observation of a previous study [Li et al., 2020] that larger models tend to converge faster. One possible explanation could be that larger models have to optimize an error surface of higher dimensionality, leading to more possible paths for gradient descent, some of which might lead to convergence more rapidly. Thus, although adding more embedding in the model results in diminishing returns in terms of predictive performance, it can lead to faster training. However, more experiments are needed to investigate this issue.

Concerning memory usage, we found that, as expected, the smaller Char model required substantially smaller amounts of memory, especially during training as evidenced by Figure 5.5. Note that, the RAM usage of the Original model is mostly due to GloVe pre-trained vectors.

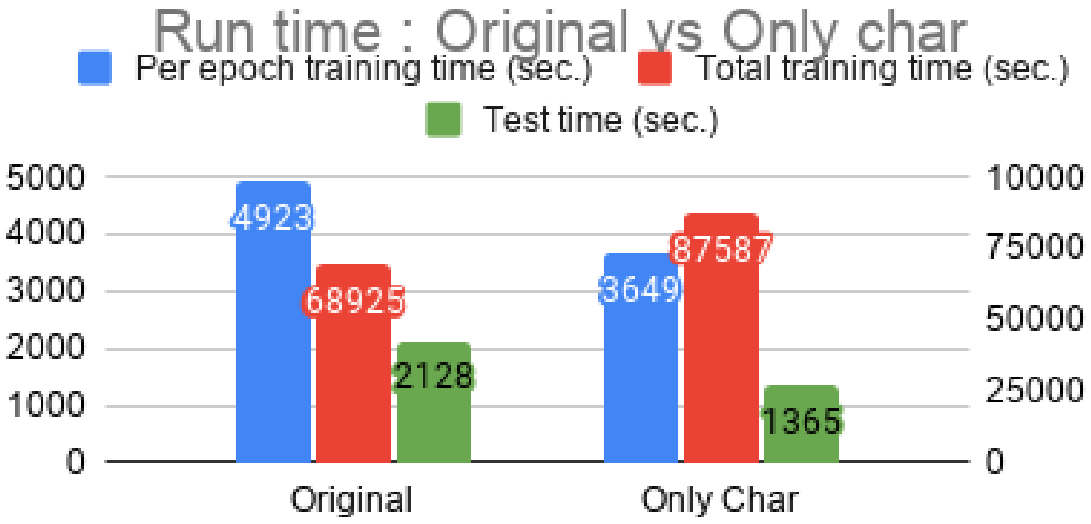


Figure 5.4: Run-time between the largest (54M weights) and smallest (677k weights) models. The total training time is associated with the right axis while the other measures are associated with the left axis.

[5] Ran on a Ryzen 5 3600X CPU and an RTX 2070 Super GPU along with 32GB of RAM

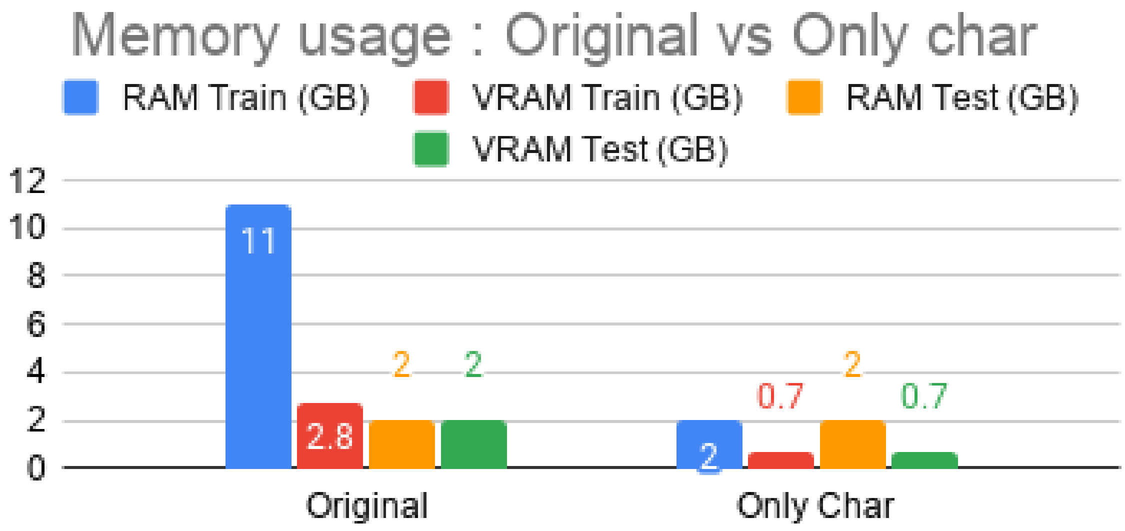


Figure 5.5: Memory usage between the largest (54M weights) and smallest (677k weights) models.

5.5.2 Results 2: Static Embeddings

We now focus on the second set of experiments, focusing our attention on static embeddings. The models concerned are from Group 2 of Table 5.1.

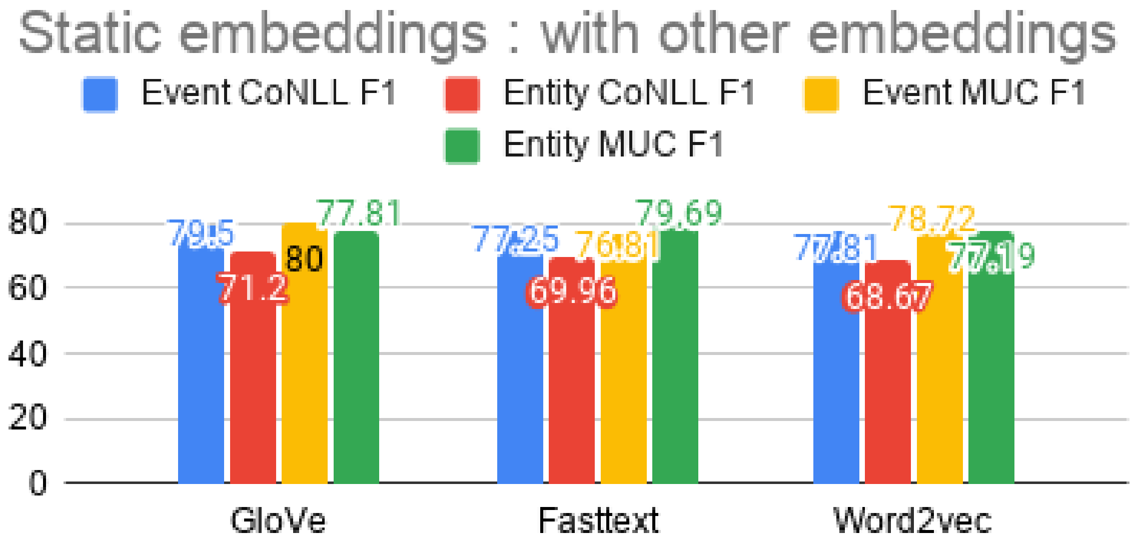


Figure 5.6: Comparing the predictive performance of static embeddings when used with other embeddings (ELMo and Character)

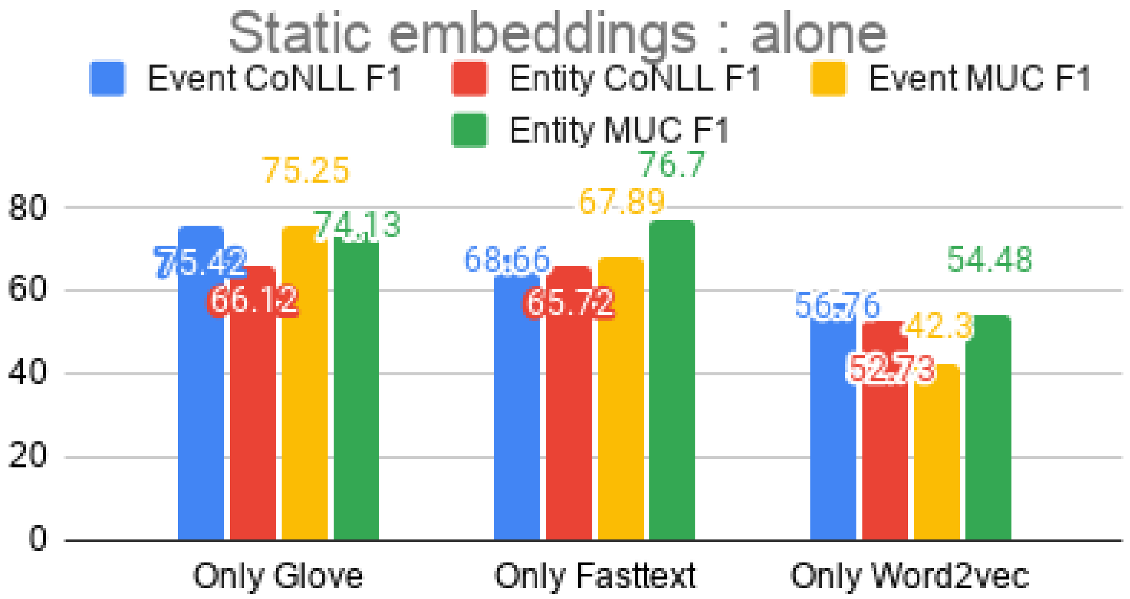


Figure 5.7: Comparing the predictive performance of static embeddings when used alone

First, we varied the static embedding (GloVe, Word2Vec, FastText), while keeping the same contextual embedding and character embedding as in the Original model. It can be seen in Figure 5.6 that, when used with other embeddings (contextual and character), all static embeddings show comparable performance. The average performance ranges from 77.12 (GloVe) to 75.59 (Word2Vec). This corroborates with our earlier findings of Section 5.5.1 whereby the model with only contextual and character embeddings, i.e. Contextual/char, achieved comparable performance to the Original (static/contextual/char) model, indicating that the specific static embedding chosen contributed only marginally to the model's performance.

However, when used alone (see Figure 5.7), we see a drastic difference in performance between them; with the average performance ranging from 72.73 (GloVe) to 51.56 (Word2Vec).

Thus, it is only when studied alone that static embeddings show their differences. Once we isolate static embeddings, we see GloVe works best for EvCR. However, for EnCR, the FastText model shows a significantly higher MUC. The better performance of GloVe and FastText with respect to word2vec can be explained by their construction. Compared to Word2Vec, GloVe takes word co-occurrence information into account. If coreferent event mentions are more likely to share co-occurring words, it would explain parts of the performance gain. FastText also outperforms Word2Vec; here the difference is that FastText takes sub-word information into account which can be advantageous for coreferent entity mentions. E.g. in Figure 2.1, "Korea" and "Korean" have similar sub-word information.

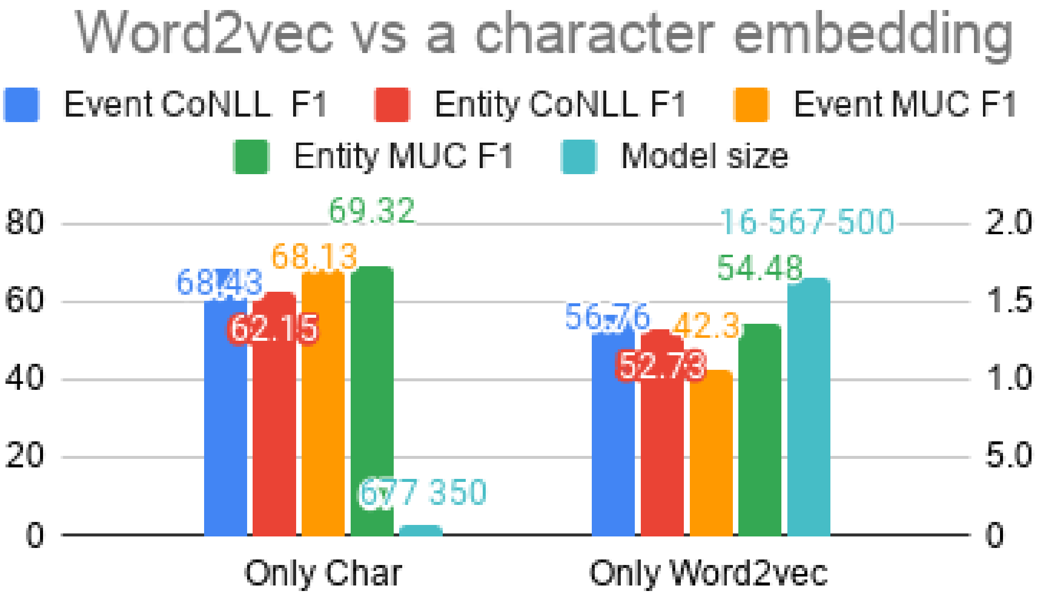


Figure 5.8: Comparing the predictive performance of solely Word2Vec vs solely a character embedding

What is most surprising is that Word2Vec is significantly outperformed by a simple character embedding as we can see in Figure 5.8. Moreover, in terms of dimension, Word2Vec has 300 and the character embedding has 50. Thus, the resulting model is not only more accurate but also ∼24 times smaller (Figure 5.8). This could indicate that the internal structure of a word (char embedding) contains more information about possible coreferences than its usual entourage (Word2Vec).

5.5.3 Results 3: Contextual Embeddings

We now focus on the third set of experiments about contextual embeddings. The models concerned are from Group 3 of Table 5.1.

Similarly to the previous Section, we present the performance of different contextual embeddings when used in tandem with the static (GloVe) and character embedding of the original model (Figure 5.9) or when used alone (Figure 5.10). We see the same as in the previous Section, i.e. the difference in performance between the contextual embeddings is clearer when they are used alone versus when they are used with GloVe and a character embedding. Thus, we will only focus on Figure 5.10 which better represents the differences between ELMo, BERT, and GPT-2.

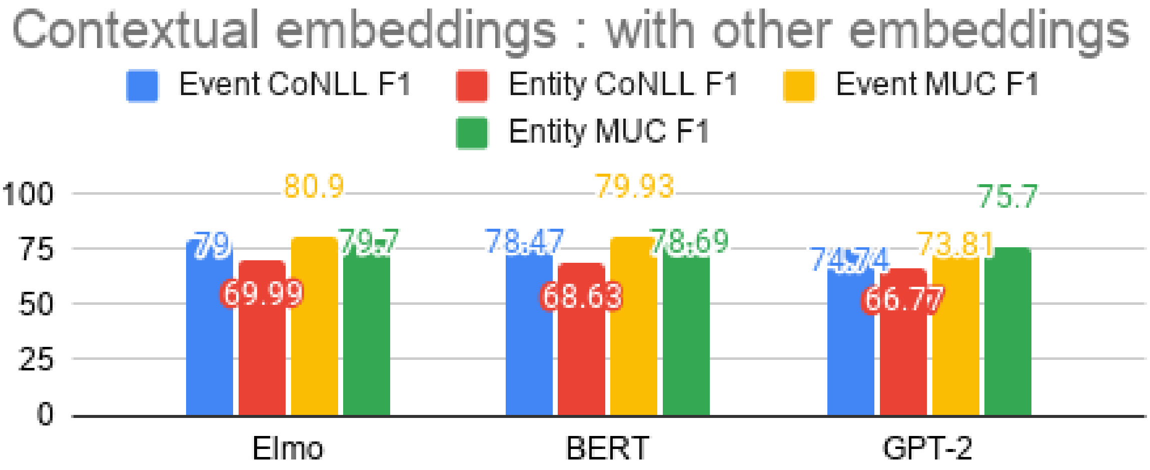


Figure 5.9: Comparing the predictive performance of contextual embeddings when used with other embeddings (GloVe and Character embedding)

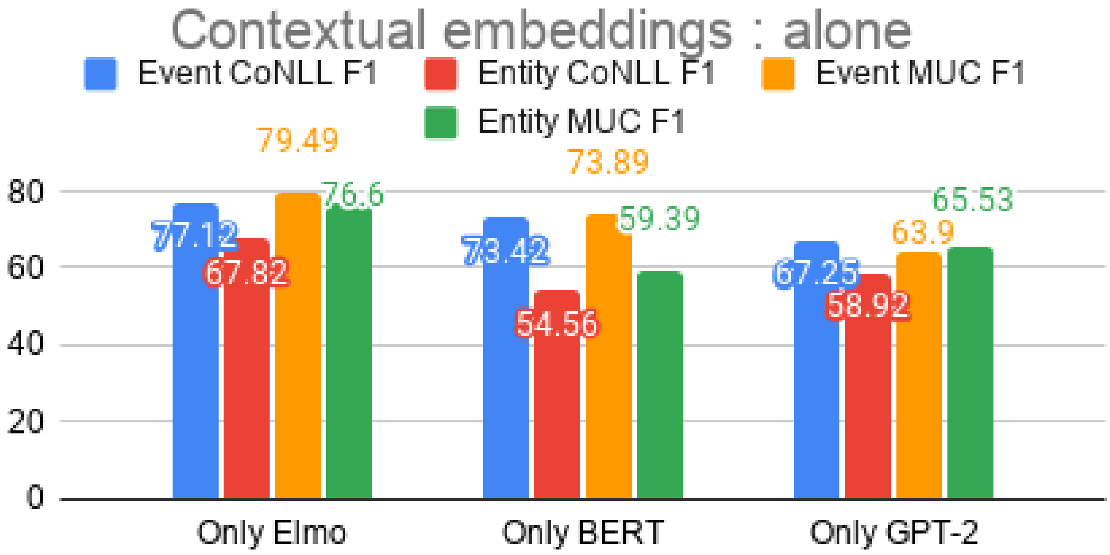


Figure 5.10: Comparing the predictive performance of contextual embeddings when used alone

A first observation is that BERT both outperforms and is outperformed by GPT-2 on both tasks. Specifically, BERT performs better in EvCR while GPT-2 performs better in EnCR.

A second observation is that ELMo clearly outperforms GPT-2 and BERT on both tasks. This result contradicts Joshi's model that found that BERT greatly outperforms ELMo on EnCR (+11.5 F1 on the GAP benchmark) [Joshi et al., 2019]. Such disparity may be indicative of differences in the model and dataset. The aforementioned model uses a span ranking approach that asks, for each mention, which is the most likely antecedent [Joshi et al., 2019]. This implicitly produces a tree that clusters coreferent mentions. Such a method only takes local information between two mentions into account while the method used in Barhom's model uses global information between two entity clusters and related event clusters [Barhom et al., 2019]. Moreover, ECB+ or EventCorefBank+ is an EvCR dataset first and foremost and only defines EnCR to support EvCR; you could argue that the EnCR tasks are more about argument than entities. GAP on the other hand is a corpus of ambiguous pronoun-name pairs [Joshi et al., 2019].

Thus, while an EnCR task is defined by both datasets, they are significantly different. We argue that both the task definition and the use of global versus local information play a major role in the disparity between the performance reported by Joshi's study and our study. Further confirming these findings would require evaluating Barhom's model on GAP and Joshi's on ECB+. However, these models are not interchangeable because the datasets and the task they define differs.

5.6 Conclusion

We used the state-of-the-art in EvCR [Barhom et al., 2019] as a framework to investigate the complexity-performance trade-off and compare the predictive performance of word embeddings across and within the three families.

We observed that the smallest model using solely a character embedding yielded 86% of the performance of the original (largest) model (using Elmo, GloVe and a character embeddings) despite being only 1.2% of its size. In fact, that smallest model achieves similar performance (68.43 F1) to the previous state-of-the-art in EvCR (69 F1) [Kenyon-Dean et al., 2018].

Paradoxically, we found that the largest model converged faster during training (by 21% in overall run-time) as it took only 14 epochs vs 24 for the character model. Overall, we found size and run-time to be weakly correlated.

In addition, our experiments revealed that augmenting the model with additional embeddings does not substantially improve the performance, leading to diminishing returns in term of predictive performance per embedding.

Concerning predictive performance, one of our most interesting result is that the model using solely a character embedding significantly outperformed ($\sim$+10 F1) a larger model using solely a static embedding (Word2Vec) while being radically smaller (4% of its size). Hence, while character embeddings have often been used as supplementary embeddings, they can actually compete with other embeddings' families in terms of predictive performance per size.

Finally, our experiments lead us to conclude that for the task of Event and Entity Coreference Resolution, GloVe, FastText and Elmo yielded the best predictive performance. GloVe and FastText performed best in EvCR and EnCR respectively in their family while Elmo performs best overall.

Future directions include working on other comprehensive study of embeddings in other tasks and experimenting with CR models using different embeddings for different tasks to improve performance. E.g. GloVe and FastText in EvCR and EnCR respectively.

This Chapter was published as a long paper in EMNLP 2021, which represents one of the best NLP conference world-wide. Nonetheless, we faced several challenges with this approach, including high pipeline error stemming from the extraction and subsequent coreference of events, the lack of consistent definitions and benchmarks in the literature, and the potential for poor generalization on unseen data due to supervised solutions. In response to these drawbacks, we shifted our focus towards unsupervised methods and broadened our scope to encompass identifying emerging trends more comprehensively, rather than just concentrating on weak signals. As a result, the Event Coreference Resolution task was no longer part of the final methodology. Hence, we had to refine our methodology and change the scope of our investigation. However, the insights and contributions from this Chapter on coreference resolution remain valuable.

6 Hierarchical Topic Modelling Over Time

6.1 Preamble

In this Section, we introduce our second paper, which delves into Hierarchical Topic Modelling Over Time (HTMOT), a novel approach that serves as the foundation of our methodology for weak signal extraction and emerging trend analysis. By integrating both hierarchical and temporal dimensions into topic modeling, we can effectively extract more in-depth and nuanced insights from textual data, which is crucial for supporting well-informed decision-making across a range of applications, including environmental scanning.

Our paper explores recent developments in topic modeling techniques that focus on either hierarchy or temporality. Modeling temporality bolsters the precision of topics by differentiating them based on distinct events, while modeling hierarchy facilitates a more comprehensive understanding of a corpus by revealing overarching topics and related sub-topics. Despite the advantages of each approach, no existing models combine hierarchy and temporality, which could yield significant benefits in a variety of applications.

To bridge this gap, we introduce HTMOT, an innovative method that merges hierarchy and temporality into a unified, coherent model. We assess the performance of our approach using a corpus of news articles and the Word Intrusion task, demonstrating the efficacy of our model in generating topics that seamlessly blend hierarchical structure and temporal aspects. Furthermore, our proposed Gibbs sampling implementation exhibits competitive performance when compared to previous state-of-the-art methods, specifically Stochastic Variational Inference, emphasizing the potential of HTMOT in extracting valuable insights from text data.

6.2 Introduction

In the field of natural language processing (NLP), numerous methods for extracting topics from a corpus have been proposed over the years [Alghamdi and Alfalqi, 2015, Barde and Bainwad, 2017]. While the seminal Latent Dirichlet Allocation (LDA) algorithm [Blei et al., 2003] paved the way for topic modeling, it lacks the ability to capture hierarchical or temporal information.

Recently, hierarchical topic models have been proposed [Paisley et al., 2015, Blei et al., 2004] that enable the extraction of topics and sub-topics organized in a tree-like structure. These models dynamically determine the appropriate number of topics and sub-topics during training and have been found to be useful in ontology learning [Zhu et al., 2017] and research idea recommendation [Wang et al., 2019].

In parallel, temporal topic models have been developed [Wang and McCallum, 2006, Nallapati et al., 2007, Song et al., 2008, Blei and Lafferty, 2006] that allow for the extraction of topics that describe events or trends occurring in a corpus. These models have been applied to tasks such as tracking trends in scientific articles [Hong et al., 2011] and events in social media [Zhou and Chen, 2013].

Combining hierarchical and temporal information in models can capture broad and detailed aspects of a corpus, benefiting applications like environment scanning [El Akrouchi et al., 2021]. Hierarchical modeling yields detailed topics and sub-topics for a comprehensive thematic understanding, while temporal modeling provides precise descriptions of events. This integration produces nuanced models for informed decision-making and deeper insights.

However, integrating temporal and hierarchical information in topic models remains a challenge [Nallapati et al., 2007, Song et al., 2008, Blei and Lafferty, 2006, Wang and McCallum, 2006]. Many temporal models have their own structure to represent time, such as time trees or time slices, which complicates the integration with a hierarchical structure [Nallapati et al., 2007, Song et al., 2008, Blei and Lafferty, 2006]. The only temporal model that does not require its own structure is ToT [Wang and McCallum, 2006], but combining time and hierarchy is still difficult due to the beta distribution used to model time lacking a known conjugate prior, making it incompatible with stochastic variational inference (SVI) used by previous hierarchical models [Wang and McCallum, 2006].

Our proposed method, Hierarchical Topic Modelling Over Time (HTMOT), jointly models topic hierarchy and temporality to leverage the strengths of both dimensions and overcome the challenges associated with integrating them.

As a secondary contribution, we propose a novel implementation of Gibbs sampling based on a tree-based data structure called the *Infinite Dirichlet Tree*. This implementation is comparable to stochastic variational inference (SVI) in terms of speed. Our work provides a promising avenue for addressing the need for topic models that can incorporate both hierarchical and temporal information. [Wang and McCallum, 2006]

We performed our experiments using a corpus of 62k news articles and evaluated our method using the Word Intrusion task [Chang et al., 2009].

6.3 Methodology

We now describe our method for Hierarchical Topic Modelling Over Time (HTMOT). We begin by presenting a new type of data structure at the core of HTMOT (section 6.3.1). Next, we describe how temporality was incorporated into the hierarchy (section 6.3.2). Then, we detail our novel implementation of Gibbs sampling (section 6.3.3). Finally, we denote important differences between HTMOT and its predecessor (section 6.4.1).

6.3.1 Counting words using Infinite Dirichlet Trees

Infinite Dirichlet Trees (IDTs) are efficient tree-based data structures we developed. The name refers to the potentially infinite number of topics provided by the Dirichlet Processes, which define how they grow. The role of these trees is to model the topics, their hierarchical dependency, and temporality. Hence, these trees are optimized during the training process to serve as the final output of HTMOT.

Each node of an IDT is identified by a finite path in the tree as a sequence of node ids, starting from the root. For example, the node "root.A.B" corresponds to a sub-topic of the topic "Root.A". The nodes record word assignments (see Figure 6.1) and the timestamps of those words (associated with the source document). Thus, each node represents a topic and defines a *topic-word* and a *topic-time distribution.*

The trees also model the hierarchical distribution of topics. Words are assigned to a final topic and to all ancestors of that topic. Hence, there are two types of word assignments: "through" and "final", respectively for the ancestor topics and final topic. This creates a hierarchical dependency between the nodes and thus a *hierarchical distribution.*

We use multiple IDTs, one for the corpus and one for each document. All words in the corpus are assigned to nodes of the corpus tree. Similarly, each document has an associated document tree recording each word of that document. Hence, combining all document trees together would yield the corpus tree. For both the corpus and document trees, each node (topic) will be assigned a different number of words. Thus, nodes differ in size which creates a distribution. Hence, the corpus tree defines a *corpus-topic distribution* and each document tree defines a *document-topic distribution.*

From the foregoing discussion, we can see that the assignment of words to the different trees defines the *topic-word, topic-time, document-topic, corpus-topic, and topic-hierarchy distributions.* Hence, by simply moving words around in those trees, we can optimize all these distributions jointly. Once optimized, the trees can be used directly as output to view topics, their hierarchy, and temporality for the corpus and each document.

6.3.2 Modelling temporality

Temporality is modelled by associating topics with a beta distribution as in ToT (Topic over Time) [Wang and McCallum, 2006]. This allows us to extract topics that describe specific event in time. Mathematically, we separate topics that are lexically similar but located at different periods in time. However, applying temporality to high level topic would split them into various periods. Each of these splits would have similar sub-topics, which would lead to an unnecessary multiplication of topics. Hence, contrary to ToT, we do not apply temporality to all topics but only deep ones. For our experiments, we choose depths of 3 or more. This allows us to extract precise topics about specific events in time at the deeper levels while keeping the high level topics intact.

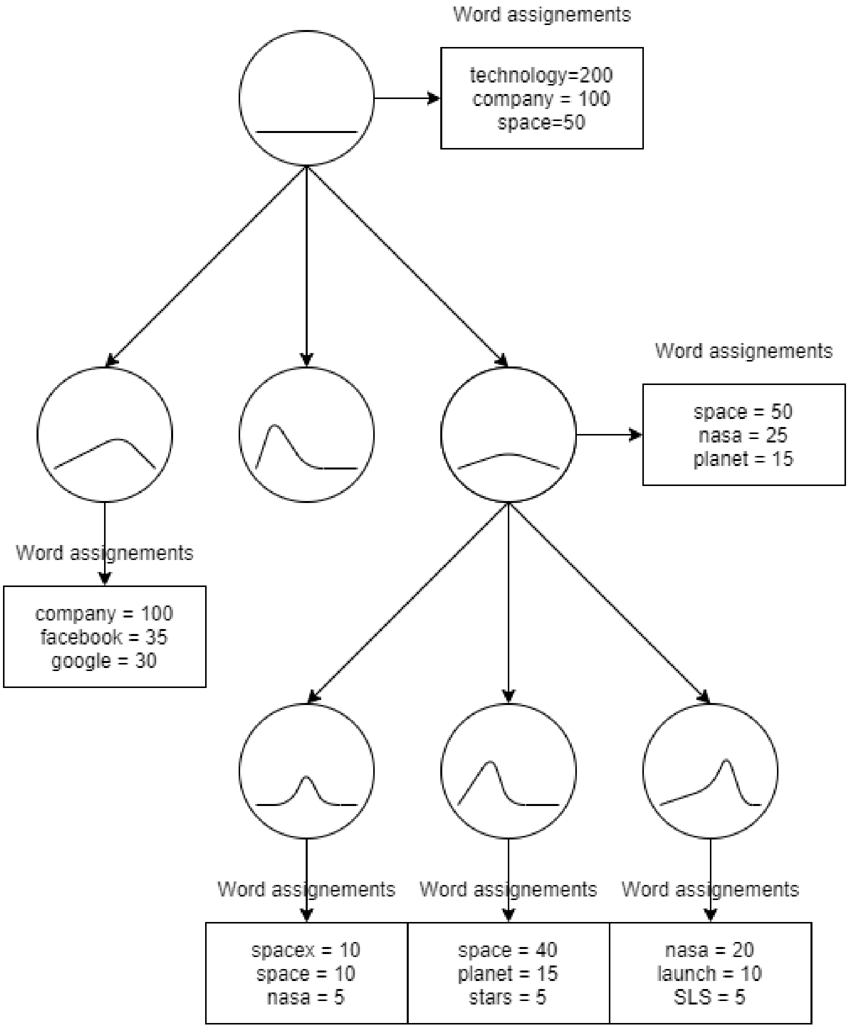


Figure 6.1: Example of an IDT with word assignments and time distribution (inside nodes).

The parameters of the beta distribution ρ_i^1 and ρ_i^2 are computed for a topic i based on the current timestamps assignments (associated with each word assignment). We used the method of the moment to estimate these parameters :

$$\rho_i^1 = \overline{t_i} * (\frac{\overline{t_i} * (1 - \overline{t_i})}{\sigma_{t_i}} - 1) \tag{6.1}$$

$$\rho_i^2 = (1 - \overline{t_i}) * (\frac{\overline{t_i} * (1 - \overline{t_i})}{\sigma_{t_i}} - 1) \tag{6.2}$$

where $\overline{t_i}$ is the empirical average timestamp assigned to topic i and σ_{t_i} is the empirical variance. These parameters are updated each time a word is assigned or unassigned to topic i.

Algorithm 1 Traditional Gibbs sampling

```
procedure ClassicGibbs(corpus)
    for N iterations do
        for each document in corpus do
            for each word in document do
                Sample topic-word assignment
                Sample topic-time
                Sample word-topic
                Sample document-topic
                Sample corpus-topic
                Sample hierarchy-topic
            end for
        end for
    end for
    Return solution
end procedure
```

6.3.3 Training HTMOT using Gibbs sampling

Two methods are commonly used for training topic models : Gibbs sampling and Stochastic Variational inference (SVI). Gibbs sampling is asymptotically exact[1] unlike SVI [Blei et al., 2017] This means that Gibbs sampling is more precise which is particularly significant when the dataset is small such as for small topics which are represented by a smaller subset of the data. However, classical implementations of Gibbs sampling are prohibitively slow as they require sampling from all distributions (see algorithm 2).

Nevertheless, in the context of topic modelling, we can avoid this issue and greatly speed up the process [Xiao and Stibor, 2010]. Specifically, it is possible to only draw from the word-topic assignment distribution. This requires the construction of a data structure tailored to the model to implicitly represent the other distributions. This is the role played by our Infinite Dirichlet Trees.

As stated in Section 6.3.1, IDTs model the aforementioned distributions based on how words are assigned to them. Hence, simply by iteratively re-arranging the words in the trees, we are implicitly optimizing these distributions. This is the key to speed up the Gibbs sampling process and represents our secondary contribution.

Hence, our training procedure consists essentially of three steps (see Figure 6.2). For each word of each document in the corpus :

1. Unassign the word from its current topic (and its ancestors) in the corpus and associated document tree.

[1] I.e. it can produce exact samples of the target distribution as the number of iterations approaches infinity. On the other hand, SVI uses a kind of Stochastic Gradient Descent which is not guaranteed to converge to a global optimum and thus will likely not produce exact samples of the target distribution.

2. Draw a random topic assignment z for that word from the word-topic assignment distribution.

3. Re-assign the word to the chosen topic z (and its ancestors) in the corpus tree and associated document tree.

This procedure is repeated until convergence which is assessed by observing empirically during training that the curves representing the frequency of each topic have flattened. Note that, changing a word's topic assignment will also update the estimated time parameters of the affected topics (equation 6.1). The initialization procedure of our algorithm is similar expect that it ignores the first step as all words starts unassigned.

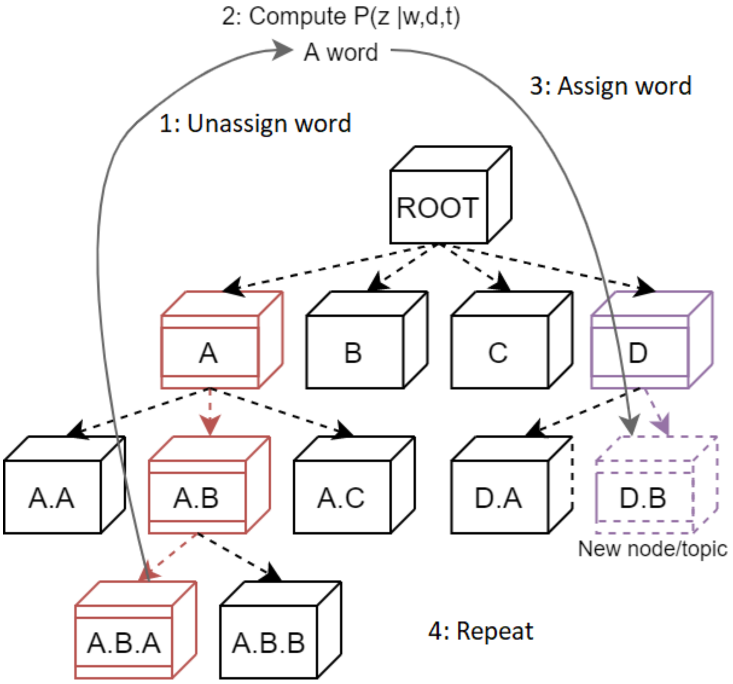


Figure 6.2: Gibbs sampling with Infinite Dirichlet Trees. Repeat for each word of each document until convergence.

6.4 Sampling topic-word assignments (paths in the trees)

We will now explain the procedure behind sampling from the topic-word assignment distribution. When drawing a topic assignment for a word we have three possible outcomes: (1) We draw a node/topic from the associated document tree, (2) We draw a node/topic from the corpus tree or (3) We create a new node/topic.

Formally, given a word w with timestamp t in document d, we wish to draw a new topic assignment z. As stated in Section 6.3.1, topics are identified as a sequence of node ids. Thus, we iteratively draw the random sequence $z_{0,L} = (z_0, ..., z_L)$. The length L of this sequence is decided by sampling a Bernoulli distribution in-between the sampling of each z_j.

Hence each z_j is sampled in a two step process. First, we decide how the topic will be drawn using a multinomial distribution :

$$Case \sim$$

$$\begin{cases} (1)\ Create\ a\ new\ topic\ with\ probability\ \frac{\beta}{\beta+n_w} * \frac{\alpha}{\alpha+n_d}\ or\ if\ z_{0,j-1}\ has\ no\ subtopics & (6.3) \\ (2)\ Draw\ from\ the\ document\ tree\ with\ probability\ \frac{n_d}{\alpha+n_d} & (6.4) \\ (3)\ Draw\ from\ the\ corpus\ tree\ with\ probability\ \frac{n_w}{\beta+n_w} * (\frac{\alpha}{\alpha+n_d}) & (6.5) \end{cases}$$

Unless case (1) is chosen, a topic needs to be drawn from one of the trees, then we sample a topic using the following distribution :

$$z_j|w,d,t \sim$$

$$\begin{cases} \sum_k \frac{\beta_k(t)*(A(k|d)+\epsilon)*(A(k|w)+\phi)*\delta_k}{(A(k)+(\phi*V))*n_d}\ if\ case\ (2) & (6.6) \\ \sum_k \frac{\beta_k(t)*(A(k|w)+\phi)*\delta_k}{n_w}\ if\ case\ (3) & (6.7) \end{cases}$$

Variable	**Description**
n	# words in the corpus
n_d	# words in the corpus that are part of document d
n_w	# words in the corpus that are instantiations of the word w
V	Vocabulary length
$A(k\|w)$	# words w assigned to topic $(z_{0,j-1}, k)$ or its descendants (corpus tree information)
$A(k\|d)$	# words in document d assigned to topic $(z_{0,j-1}, k)$ or its descendants (document tree information)
$A(k)$	# words assigned to topic $(z_{0,j-1}, k)$ or its descendants
β_k	Probability density function of the beta distribution with parameter ρ_k^1 and ρ_k^2 associated with topic $(z_{0,j-1}, k)$
$\epsilon, \phi, \beta, \alpha$	Priors for the Dirichlet distributions and processes (more details are provided in the parameter Section)

Table 6.1: Descriptions of variables for equations 6.6 to 6.3

Note that sampling a node from the corpus tree can lead to the creation of a new node in the associated document tree if that node does not already exist. However, when creating an entirely new node, it is created in both trees (corpus tree and associated document tree).

Once a topic z_j is drawn, we draw from a Bernoulli with parameter p to decide if we stop or go deeper in the tree. p is computed with the following formula :

$$p_{\epsilon,\phi}(t,w,d) = \frac{P_{\epsilon,\phi}(t,w,d) + \theta_1}{N_\phi + \theta_1 + \theta_2 + C_{\epsilon,\phi}(t,w,d) + P_{\epsilon,\phi}(t,w,d)} \tag{6.8}$$

.

$$P_{\epsilon,\phi}(t,w,d) = \frac{\beta_j(t) * (A^*(z_{0,j}|w) + \phi) * (A^*(z_{0,j}|d) + \epsilon)}{A^*(z_{0,j}) + (\phi * V)} \tag{6.9}$$

$$N_\phi = \frac{\phi * \epsilon}{\phi * V} \tag{6.10}$$

$$C_{\epsilon,\phi}(t,w,d) = \sum_k \frac{\beta_k(t) * (A(k|w) + \phi) * (A(k|d) + \epsilon)}{A(k) + (\phi * V)} \tag{6.11}$$

With $A^*(z_{0,j})$: the number of words assigned to topic $z_{0,j}$. P : the weight of the currently selected node $z_{0,j}$. C : the weight of all of the children of the selected node $z_{0,j}$. N : the weight of a potentially new child for $z_{0,j}$ and θ_1 / θ_2 : the priors for the Bernoulli distribution.

To summarize, when drawing a topic assignment for a word, we either draw from the document tree, corpus tree, or we create a new topic. Then, we draw from a Bernoulli to decide if we go deeper or not. If we do go deeper, we repeat the same process until we eventually stop. This process is then applied repeatedly too all of the words in the corpus multiple times until convergence (see Algorithm 2 for more details).

6.4.1 Comparing HTMOT vs. nHDP

The main difference between HTMOT and nHDP is their use of Gibbs sampling and SVI training procedures, respectively. However, other notable differences exist. Firstly, our HTMOT algorithm starts with all words unassigned, while nHDP uses a pre-clustering step with k-means. Secondly, we do not use a greedy algorithm to select trees for each document. Instead, the tree for each document is created automatically as the Gibbs sampler progresses. As a result, our training algorithm is simpler and easier to implement, avoiding the need for pre-clustering or greedy procedures.

6.5 Experimental setup

6.5.1 Dataset

To perform our experiments, we crawled 62k articles from the Digital Trends [2] archives from 2015 to 2020. The crawling was performed using Python with the help of the BeautifulSoup library. Digital Trends is a news website that mainly focuses on technological news but also contains general news. For all articles, we extracted the text, title, and timestamp.

The timestamps were mapped to a number between 0 and 1, which corresponds to the domain of the beta distribution used. Hence, 0 corresponds to the earliest date of a document in the corpus, and 1 corresponds to the latest.

We cleaned the data as follows. First, we removed common editor's sentences such as "*we strive to help our readers....*" to remove noise from the data. Then, we relied on Spacy's NER and POS to filter relevant tokens. Specifically, we kept specific kinds of entities (Person, Norp, Fac, Org, Gpe, Loc, Product, Event, Work_Of_Art, Law, Language) and POS elements (ADJ, NOUN, VERB, INTJ, ADV). Finally, lemmatization was also applied.

[2] `https://www.digitaltrends.com/`.

Algorithm 2 Gibbs sampling using Infinite Dirichlet Trees

```
procedure IDTGibbs($corpus$)
    $IDT$ = Empty Infinite Dirichlet Tree (only a root)
    $N$ = Number of iterations
    for $iteration$ in $range(N)$ do
        for each $document$ in $corpus$ do
            for each $token$ in $document$ do
                sampleTopic($token$,$IDT$,$iteration$)
            end for
        end for
    end for
    Return IDT
end procedure

procedure sampleTopic($token$,$IDT$,$iteration$)
    z = (root)
    $y$ = current topic assignment for $token$ if it was assigned to a topic before
    Unassign $token$ from $y$ if $y$ is defined

    while true do
        Sample topic $t$ from topic-word distribution (see 6.6)
        if $t$ is a new topic (see 6.6) then
            if $iteration < SGI$ (see 6.5.2) then
                Create a new topic $t$ in $IDT$
                Append $t$ to $z$
            end if
            Exit the loop
        end if
        Append $t$ to $z$

        if $size(z) < SM$ (see 6.5.2) then
            Exit the loop
        end if

        Draw $continue$ from a Bernoulli distribution with parameter $p$ (see 6.8)
        if $continue == 0$ then
            Exit the loop
        end if
    end while

    Assign the $token$ to $z$
    Update topic-time distribution estimation for all topics in the path of $z$ and $y$ (see 6.3.2)

    for each $topic$ in $IDT$ do
        if size($topic$) $< CM$ for more than $TTL$ iterations (see 6.5.2) then
            Delete $topic$
        end if
    end for
end procedure
```

A good pre-processing procedure is essential for the interpretability of topics, as shown in [Martin and Johnson, 2015]. Hence, our extraction of named entities aims to enhance the topics' interpretability by showing actors in the topic such as personalities and companies. The training algorithm will not discriminate between words and entities, but the visualization interface does. This means that a topic is no longer displayed as a simple list of words but is instead represented by a list of words and a list of entities.

6.5.2 Parameters

Many parameters control the behavior of our model; this Section will describe each of them.

First, we have the Infinite Dirichlet Trees parameters. α : the rate at which we create new topics in the document trees. β : the rate at which we create new topics in the corpus tree. θ : how likely we are to create deeper sub topics.

Second, we have parameters that regulate the growth of the trees. These help speed up the algorithm and keep memory usage to a minimum. CM (Critical Mass) : the minimum valid size of a topic; only valid topics are part of the final output. SM (Splitting Mass) : the minimum size of a topic before it can create sub-topics. Both are defined as a percentage of the total number of words in the corpus. TTL (Time To Live) : how many pass through the corpus before destroying a non-valid node. Nodes are also destroyed when they become empty.

Third, we have the Dirichlet prior parameters as in the traditional LDA model. ϕ : the prior for the topic-word distribution. ϵ : the prior for the corpus and document-topic distributions.

Finally, we have training parameters. Iterations : how many batches we will go through during training. SGI (Stop Growth Iteration) : a point at which node new nodes won't be created. Set SGI $<$ Iterations to ensure that the last topic to be created has time to converge.

Table 6.2 defines the value of each parameter used to perform our experiments.

Parameter	Value
α	0.00005
β	0.0002
θ	0.25
Critical Mass (CM)	0.0005
Splitting Mass (SM)	0.005
Time To Live (TTL)	2
ϕ	0.1
ϵ	1
Iterations	4500
Batch size	500

Table 6.2: Parameters used for our model

6.6 Results and Discussion

We now present our results, starting with a statistical analysis of the training behavior of HTMOT. Then, we will discuss the results of the Word Intrusion task, its drawbacks, and directions for future topic modeling evaluation methods. Finally, we will examine the various extracted topics qualitatively.

6.6.1 Convergence rate, training speed, and algorithmic complexity

To assess the convergence of our method during training, we looked at the frequency of depth 1 topic over time. As these frequencies stabilize, it indicates that the model has converged. Since hierarchical topic models extract hundreds of topics, it is not reasonable to observe the convergence of each topic.

Our experiments revealed that the convergence rate of our training algorithm is sub-linear with respect to the dataset size. Using a dataset ten times smaller leads to a halving of the time to convergence. However, new topics created during training can perturb this convergence, which is prevented by the SGI parameter (see Section 6.5.2).

To compare training times, we disabled HTMOT's temporal modeling to ensure a fair comparison with nHDP, which lacks a temporal component. Our sampler analyzes 135k documents per hour, while nHDP's SVI analyzes roughly 90k articles per hour, based on figures reported in [Paisley et al., 2015]. While hardware and programming language used differ, Gibbs sampling is generally understood to be prohibitively slow compared to SVI. Our results demonstrate that our Gibbs sampling implementation can be used effectively for large datasets. The algorithmic complexity is linear with respect to the dataset size, but the depth of topic trees and growth and regulating parameters for the IDTs can greatly impact performance. Nonetheless, as mentioned, the convergence rate is sub-linear.

6.6.2 Results of the Word Intrusion Task

We evaluated our model using the automated Word Intrusion task, testing the quality of all topics by replicating the original study[Lau et al., 2014] (see Chapter 8 for details). Unlike the classical task, we selected intruder words only from sibling topics, making the task more challenging as deeper topics tend to be more lexically related to their siblings. This is important as it helps ensure topic distinctiveness. For example, when selecting an intruder word for "astronomy", we chose from its sibling topics like "astronaut", making the chosen intruder semantically closer to the target topic (see Figure 6.3 for illustration). This approach provides a more robust evaluation of topic quality.

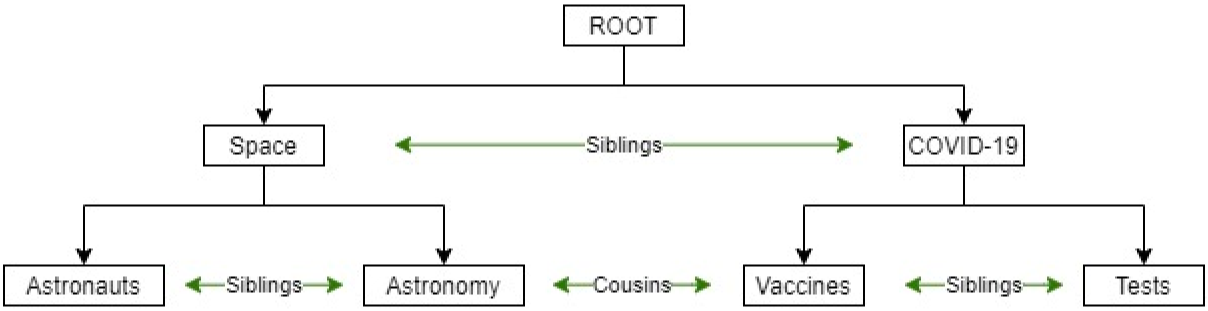


Figure 6.3: Example of a topic tree with cousins and siblings.

By applying the Automated Word Intrusion task to every topic extracted, we observed an accuracy of 98% which is similar to LDA's performance [Chang et al., 2009]. This demonstrates that HTMOT provides topics of similar quality with the added benefit of modeling temporality and hierarchy. Chapter 8 will delve into more details about the word intrusion task and its performance.

6.6.3 A qualitative examination of the resulting topics

In Figure 6.4, our model's ability to extract atomic events at the deeper level of the tree is demonstrated through the well-localized time distribution of the three sub-topics under "astronauts". These sub-topics, namely the historic test launch of the SpaceX Dragon capsule, the Crew 1 launch, and the Crew 3 launch were mostly interpreted from top documents due to their depth, making it difficult to interpret based on top words. The timing of these events matched their associated time distribution, occurring in May 2020, November 2020, and November 2021 respectively. The model missed the Crew 2 launch event, which may be related to the reduced output of digital trends news during that period, as shown in Figure 6.5.

6.7 Conclusion

We have proposed a new model for topic modeling capable of modeling hierarchy and time jointly. Through examples, we have demonstrated how combining hierarchy and temporality provides us with a more fine-grained understanding of a corpus through detailed sub-topics which can represent specific events. Moreover, we developed a novel implementation of Gibbs sampling for hierarchical topic models. This implementation provides a fast alternative to SVI that makes Gibbs sampling a viable solution for training such complex models. Moreover, we have shown how extracting entities can help interpret and understand topics at a deeper level.

Through the rest of this thesis, this Chapter will serve as the core of our methodology. Hence, all subsequent Chapter will aim at improving or evaluating the capabilities of HTMOT for the purposes of trend and weak signal detection.

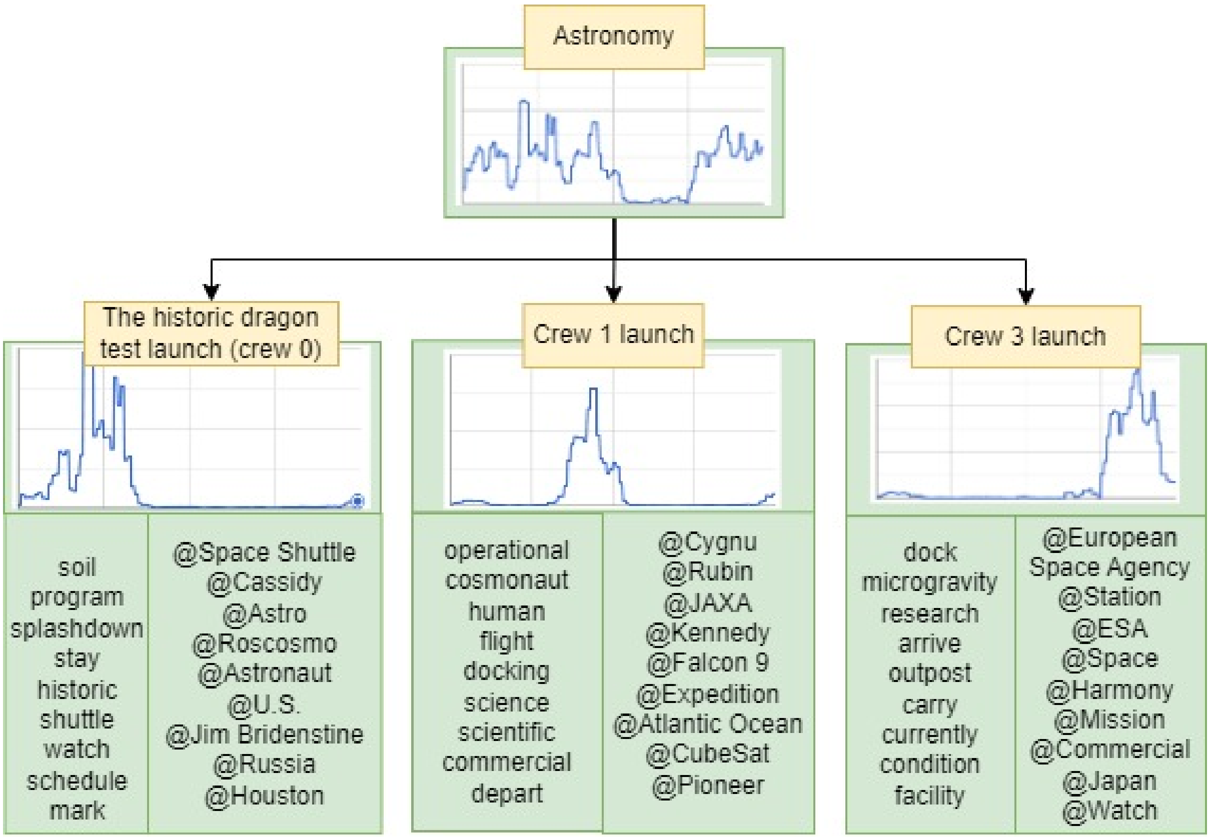


Figure 6.4: Examples of depth 3 topics that are well localized in time

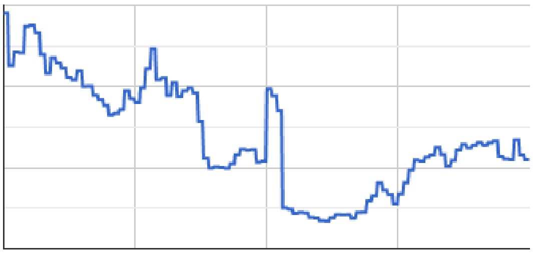


Figure 6.5: Number of articles published by Digital Trends over the years 2020 and 2021. We can see a sharp decline at the beginning of the year 2021 (middle of the graph)

7 Topic Tracking Using Word Embeddings

7.1 Preamble

In this Section, we introduce our third article, which centers around topic tracking, a key component of our methodology that allows us to understand the evolution of topics over time and gain deeper insights into emerging trends. By monitoring how topics change and develop, businesses can better adapt their strategies to stay ahead of the curve.

Existing topic tracking methods primarily rely on lexical information by matching word usage patterns. However, no studies have yet investigated the potential benefits of using semantic information for tracking topics. Therefore, we delve into an innovative semantic-based approach that leverages word embeddings to capture the underlying meanings and relationships among words.

Our findings reveal that the semantic-based method for topic tracking performs on par with the traditional lexical approach, albeit with different types of errors. This suggests that combining both methods could complement each other, resulting in a more robust and comprehensive topic tracking system. Furthermore, our paper also highlights the inherent challenges associated with tracking topics in hierarchical topic models, emphasizing the need for innovative approaches to address these complexities.

7.2 Introduction

Topics extracted through topic models can be tracked over time to understand their evolution or discover emerging ones. Hence, the goal of topic tracking is to link instances of the same topic that have been extracted at different time periods. Several methods for tracking topics have been proposed in the past [AlSumait et al., 2008, Fan et al., 2021, Zhu et al., 2016, Xu et al., 2019, Liu et al., 2020a]. These methods use measures such as the JS divergence [Zhu et al., 2016, Xu et al., 2019, Liu et al., 2020a] or online topic models [AlSumait et al., 2008, Fan et al., 2021] which rely on lexical information to track topics across time.

However, no studies have ever experimented with using semantic information to track topics over time. Intuitively, semantic-based approaches could be promising as they do not rely on simple surface form and can capture concepts such as synonymy. For example, given the topic of "AI", across time we could observe that the term "Machine Learning" has become more popular than "AI". However, a lexical approach to topic tracking would not be able to handle such lexical drift and relate those words over time. Conversely, such lexical variation would have been captured by a semantic approach. Moreover, topic-word distributions are unstable across multiple runs [Agrawal et al., 2018], i.e. the resulting top words of a topic tend to change significantly. This entails that the lexical information we rely upon to track topics is also unstable even if the overall semantic of the topic remains the same. Thus, a semantic-based approach may be more robust.

Hence, our work aims at investigating the use of semantic information for topic tracking and its comparison against lexical information. Therefore, as our main contribution, we propose a novel semantic-based topic tracking method (SD) based on word embeddings. As an ancillary contribution, we study the challenges of topic tracking in the context of hierarchical topic modeling.

7.3 Methodology

In this Section, we will present our methodology for topic tracking. We will start by describing our corpus and topic extraction method. Next, we will define our SD measure. Finally, we will present the topic tracking algorithm.

7.3.1 Topic extraction

To perform our experiments, we crawled 10k articles from the Digital Trends [1] archives from 2019 to 2020. This news website is mainly focused on technological news with topics such as hardware, space exploration, and COVID-19. For all articles, we extracted the text, title, category, and timestamp. We pre-possessed the corpus as described in Chapter 6.

To extract topics hierarchies (see Figure 7.1), we used the HTMOT topic model proposed in Chapter 6. The extracted topics are represented by a list of words and a list of entities extracted in the corpus through Named Entity Recognition as described in Chapter 6.

In accordance with Chapter 6, we only focus on the first and second levels for the extracted topics. Specifically, the authors observe that deeper topics become more esoteric making them harder to understand by annotators representing a general audience. Consequently, this makes it difficult to assess the correctness of tracked topics at deeper levels of the topic tree.

[1] https://www.digitaltrends.com/

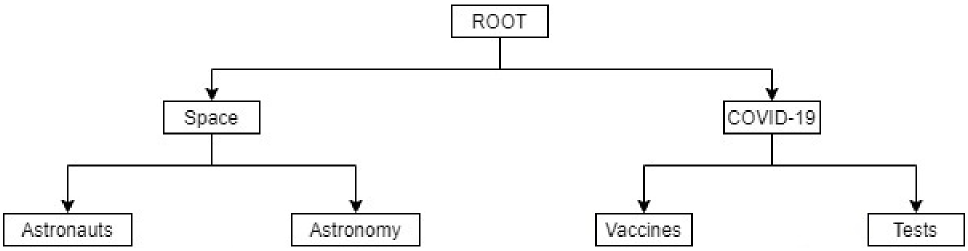


Figure 7.1: Example of a topic hierarchy

7.3.2 Proposed Semantic Divergence measure

We will now describe our novel topic tracking method, which departs from the JS divergence traditionally applied in previous studies. Our measure called "Semantic Divergence" (SD) uses word embeddings to measure the distance between topics. Each topic will be assigned an embedding as the sum of the embeddings of the top words in that topic weighted by their probability. Then, the distance between the two topics is computed as the cosine distance of their respective embedding. We will use FastText as the word embedding. FastText helps with rare and out-of-vocabulary words [Bojanowski et al., 2017]. This is essential considering our pre-processing step includes lemmatization which may produce incorrectly spelled words. Hence the embedding of a topic is defined as follows :

$$emb(t) = \sum_{(w,p)\in t} p * FastText(w) \tag{7.1}$$

And our Semantic Divergence measure between two topics is defined as :

$$SD(t_1, t_2) = 1 - \frac{emb(t_1) * emb(t_2)}{||emb(t_1)||_2 * ||emb(t_2)||_2} \tag{7.2}$$

where w is a word in a topic t and p is the probability of that word as defined by the topic t.

7.3.3 Topic Tracking Algorithm

Finally, to track topics across time we applied HTMOT to our corpus. For each year (2019 and 2020), we obtained a corresponding topic tree. Then, we computed the distance between every topic across both years using either JS or SD. To do this we used the top 100 words and top 15 entities to represent each topic. Subsequently, we ranked all computed pairs of topics and then iteratively selected the most similar pairs (lowest SD or JS score) such that each topic is paired only once. Finally, we used a pre-defined threshold to remove pairs with poor scores.

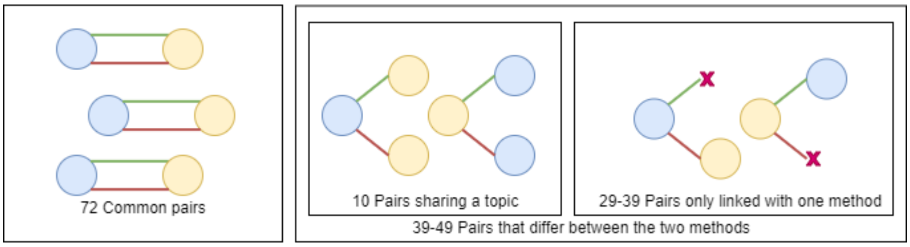

72 Common pairs
10 Pairs sharing a topic
29-39 Pairs only linked with one method
39-49 Pairs that differ between the two methods

Figure 7.2: The pairs extracted by both methods can be grouped into three categories. The circle represents topics and their color represents years (2019 blue; 2020 yellow). The link color represents the method used (JS red; SD green). The three categories are 1) The pairs extracted by both methods (72). 2) The pairs that differ but share a topic (10) E.g. JS extracted the pair 4-D while SD extracted 4-E. 3) The pairs of topics that were only linked with one method (29-39).

Note that our approach does not take into account structural information. Indeed, tracking topics in the context of hierarchical topic modeling presents another interesting challenge: there exist many possible resulting trees that are equally correct. In one run, we may extract the topic of space whose sub-topics can be grouped into space exploration and astronomy. Conversely, in another run, we may extract space exploration and astronomy as separate topics with their own sub-topics. Hence, it is difficult to leverage the structural information contained in the topic trees to track topics as it cannot be expected to respect a specific conceptual taxonomy.

7.4 Results : JS vs SD

In this Section, we will discuss how our semantic-based method compares with respect to the traditional lexical-based method.

First, we studied the overlap between the two methods, i.e. the number of pairs extracted by both. We discovered that 111 pairs were extracted with JS with a threshold of <0.4, while 121 pairs were extracted with SD with a threshold of <0.1. These thresholds were set through empirical observation and may depend on the dataset used. These 111-121 pairs can be grouped into three categories (see Figure 7.2). 72 pairs were the same between the two methods (60-65% of the total pairs). For example, topics such as space and video games were easily paired across both years by both methods. This already indicates that our SD method is able to pair topics across time with performance similar to JS. This leaves 39-49 pairs that are different across the two methods (35-40% of the total pairs) which we can evaluate. Out of those different pairs, we notice that in most cases one method (e.g. SD) would track/link a topic pair across both years, while the other method (e.g. JS) did not as the best possible pair was above the threshold. We are then left with 10 different pairs that can themselves be paired according to which 2019 or 2020 topic they share (see Figure 7.2).

To compare the performance of the two tracking methods, we decided to use a survey comparing these 10 pairs of topics extracted by both JS and SD. Precisely, for each question, given an initial topic, annotators were shown the JS and SD pairing and asked which is better. Additionally, we also asked annotators to provide a confidence score on a scale from 1 to 5. In total, we received 38 answers coming from a small online community focused on answering surveys[2]. The survey can be found on GitHub [3].

Looking at the survey results (table 7.1), it can be seen that SD slightly outperforms JS with 54% of annotators preferring the former to the latter. Moreover, we also note that the annotators were confident in their evaluation, with an average confidence score of 3.3. Interestingly, there is a lot of variability in the answers. Some topics were clearly better paired with one method or the other (Q3 and Q5) while for others, it wasn't as clear (Q1, Q2, and Q4).

Table 7.1: The "chose SD" column corresponds to the % of annotators that chose the SD pair as the best pair.

Questions	Chose SD	Confidence level
Q1	42.1% (22)	3.2
Q2	63.2% (14)	2.6
Q3	21.1% (30)	3.7
Q4	65.8% (13)	3.5
Q5	78.9% (8)	3.5
Average	54%	3.3

For example, Figure 7.3 corresponds to Q1. It shows how a 2019 topic has been paired with 2020 topics using JS and SD. First, we can notice that the distance recorded between the pairs is close to the threshold for both methods. Specifically, 0.29 for the JS pair and 0.09 for the SD pair (threshold = 0.4 for JS and 0.1 for SD). This makes sense as good pairs (pairs with low JS/SD values) are extracted by both methods. Second, the 2019 topic is about social media data security. Whereas the chosen 2020 topic is about :

- Social media when paired with JS.
- Data security when paired with SD.

Hence, both pairing seems suitable, which could explain the indecisiveness of annotators. Specifically, 16 of them decided the SD pairing was better whereas 22 of them decided the JS pairing was better. Their confidence level for this question was 3.2 out of 5.

Similarly, Figure 7.4 corresponds to Q5 and shows how another 2019 topic has been paired based on the two methods. Here, the 2019 topic is about web security. Whereas the chosen 2020 topic is about :

- Data security when paired with JS.
- Web security topic when paired with SD.

[2] `https://www.reddit.com/r/SampleSize/`

[3] `https://github.com/JudicaelPoumay/TopicTrackingPaper`

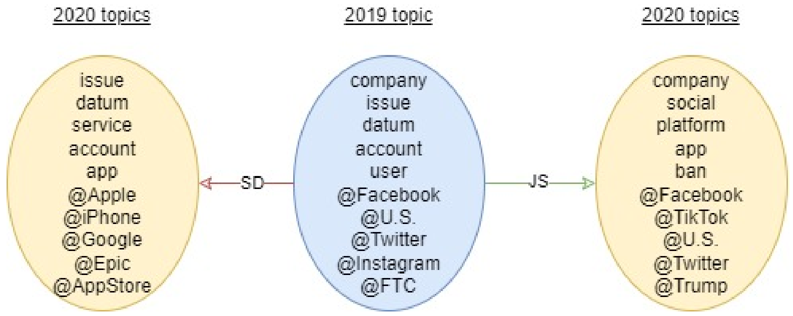


Figure 7.3: A first example of different pairing between SD and JS on the same 2019 topic.

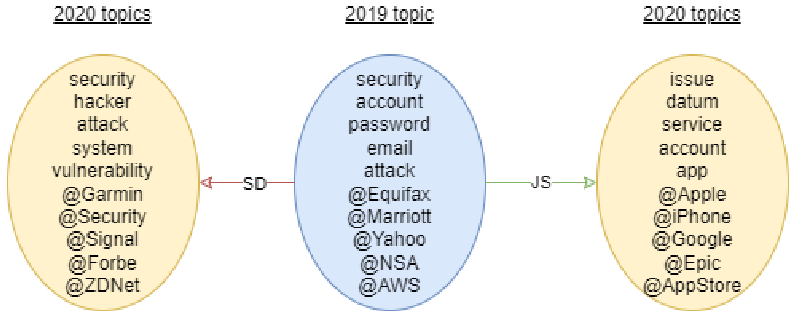


Figure 7.4: A second example of different pairing between SD and JS on the same 2019 topic.

Moreover, the topic chosen by SD is a sub-topic of the topic chosen by JS which demonstrates the difficulty in topic tracking in a hierarchical setting. Indeed, it can be difficult to differentiate a topic from its sub-topic, especially if that sub-topic dominates the others as parent topics are the sum of their sub-topics. In this case, annotators agreed more and 30 out of 38 decided the SD pair was better. Their confidence level for this question was 3.5 out of 5.

Hence, we argue that JS and SD are two fundamentally different approaches and that both have their advantages. JS is lexically driven and may work best for linking topics that tend to have a stable and precise vocabulary such as in legal documents. On the other hand, SD is driven by semantics and may be more appropriate for linking topics that have greater lexical variability. Greater lexical variability may be the result of lexical drift over time as terms change in popularity or informal texts which do not use a standard vocabulary such as tweets. Hence, we believe that SD not only competes but complements JS for topic tracking.

7.5 Conclusion

In this paper, we presented a novel semantic-based topic tracking method (SD). We showed that its performance was comparable to that of the state-of-the-art method (JS), which is lexically based. This validates our hypothesis that semantic information is valuable for tracking topics.

Moreover, we have discussed the challenges associated with tracking topics in a topic hierarchy. First, topics and their sub-topic can be difficult to differentiate, which makes topic tracking more challenging. Second, deeper topics are more esoteric and consequently, it is harder to assess the quality of their tracking. Finally, a topic hierarchy may have many equally correct arrangements which make it difficult to leverage structural information for topic tracking.

We believe that our work would benefit future studies investigating hybrid methods for topic tracking, such as by integrating lexical and semantic information.

This chapter was published as a short paper in NLDB 2022 which is further published in Lecture Notes in Computer Science, a Book published in Springer. In the context of this thesis, this chapter provides the final element of our methodologies by allowing us to understand the evolution of topics through time. This move from static analysis to a temporal one allows us to understand not only weak signals but also discover emerging trends and detect larger micro and macro trends.

Part IV

EVALUATION

A significant portion of this thesis was dedicated to developing a methodology for evaluating hierarchical topic models. As existing evaluation methods were shown to be less reliable than initially thought, we explored alternative approaches, including the word intrusion task and its automated variant. However, our investigation revealed that the word intrusion task was not well-suited for evaluating hierarchical topic models either. Indeed, hierarchical models can discover hundreds of small sub-topics in many domains. Each small sub-topic requires a high level of domain knowledge making the method unscalable and costly. Consequently, we investigated a new evaluation technique based on a labeled dataset.

We will now describe each component of the evaluation methodology in details:

- Chapter 8 will describe a critic of the Word Intrusion Task for Hierarchical Topic Models
- Chapter 9 will describe evaluating Hierarchical Topic Models using a labeled dataset

8 A Critic of the Word Intrusion Task for Hierarchical Topic Models

8.1 Preamble

In this Chapter, we present our fourth article that critiques the Word Intrusion task. While this task is considered a state-of-the-art method for evaluating topic models, its effectiveness in assessing the quality of hierarchical models has not been thoroughly investigated.

To address this gap, we conducted a series of experiments using both a human-annotated and an automated version of the Word Intrusion task on the latest hierarchical topic model. Our findings revealed that the task posed significant challenges for human annotators, particularly when evaluating deep sub-topics. In fact, the accuracy rate dropped from 79% to only 6.4% as we delved deeper into the tree.

In contrast, the automated version of the task performed remarkably well, achieving near-perfect accuracy rates even at the deepest level of the tree, with the lowest score of 95.9%. These results suggest that the Word Intrusion task may be too easy for computers to solve and may not serve as a reliable surrogate measure of human judgment.

Our findings highlight the limitations of the Word Intrusion task as an evaluation method for hierarchical topic models and underscore the need for more reliable and robust evaluation techniques. This research has significant implications for researchers and practitioners in natural language processing and related fields who rely on topic modeling to understand large text corpora.

8.2 Introduction

Several methods have been proposed for evaluating topic models, including perplexity [Blei et al., 2003], which measures how well a model predicts the probability of unseen documents, and topic coherence [Newman et al., 2010], which assesses the degree to which the words within a topic are semantically related. However, these methods have been repeatedly shown to be uncorrelated with human judgment and may not always be reliable indicators of the quality of a topic model [Chang et al., 2009, Hoyle et al., 2021, Doogan and Buntine, 2021, Bhatia et al., 2017].

To address this issue, the Word Intrusion task has been proposed as a more reliable evaluation method [Chang et al., 2009]. In this task, human annotators are asked to spot an intruder word inserted in each topic. The idea is that in good topics, annotators would easily find this intruder. In a subsequent study, the authors have shown that this task can be automated with performance that correlates with human annotators [Lau et al., 2014] . However, the effectiveness of the Word Intrusion task has not been investigated in the context of hierarchical topic models.

In this study, we address this research gap by examining the reliability of the Word Intrusion task for evaluating hierarchical topic models. We experiment with both a human-annotated and an automated version of the task and apply them to a hierarchical topic model. Our results show that :

1. The human version of the task
   - (a) is not reliable for topics that are extremely domain specific (esoteric) in nature, especially deeper topics.
   - (b) is not reliable for deep sub-topics as the intruder becomes semantically closer to the target topic.
   - (c) is not scalable for large topic trees as there can be thousands of topics to evaluate.
2. The automated version of the task
   - (a) is not correlated to human judgment once we apply it to deep topic trees since we do not observe a large drop in performance similar to the one observed with human annotators.
   - (b) is too trivial to solve for computers and does not reflect the actual performance of the model.

8.3 Methodology

To evaluate the effectiveness of the Word Intrusion task in the context of hierarchical topic models, we conducted a series of experiments using a corpus of 64k articles between 2015 and 2020 from the Digital Trends website. As we had access to a pool of annotators from the general population, we chose this corpus for our study because it includes a variety of current news topics that were easily accessible for crawling.

We used the HTMOT topic model described in Chapter 6 to extract topics from the articles, following the same preprocessing steps as described in the original paper. Specifically, we filtered relevant tokens using Spacy's Named Entity Recognition and Part-of-Speech tags and applied lemmatization.

To apply the Word Intrusion task to the hierarchical topic model, we modified the task to only consider intruder words from sibling topics rather than from any topic. This was done to make the task more challenging, as deeper topics tend to be more lexically related to their siblings. This is important as we want topics to be distinguishable from their siblings. For example, in evaluating the sub-topic "astronomy," we would select an intruder word from one of its sibling topics, such as "astronaut," rather than from an unrelated topic such as "Covid-19 vaccines" (as shown in Figure 8.1). This provides a more difficult and thus robust evaluation of topic quality.

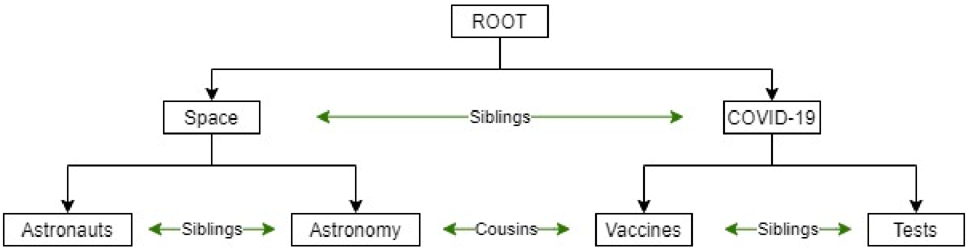


Figure 8.1: Example of a topic tree with cousins and siblings.

To collect human annotations, we conducted a survey asking annotators to select an intruder word for each of 15 topics (5 level 1 topics, 5 level 2 topics, and 5 level 3 topics). The topics were randomly selected at each depth. The survey presented 10 words for each topic, including the intruder. The annotators were recruited from an internet community involved in sharing and answering surveys [1]. A total of 91 respondents participated in the survey over the months of November and December 2022.

In addition to the human-annotated version of the Word Intrusion task, we also experimented with an automated version of the task [Lau et al., 2014]. This method involves training a ranking support vector regression (SVM^{RANK}) model using the Point-Wise Mutual Inforamtion (PMI), Normalized PMI (NPMI), and Conditional Probability (CP) between the top N words (including the intruder) as features. We reproduced this method but modified it to only compare with sibling topics and use only PMI and CP as features as it was found to be sufficient to achieve high performance.

Specifically, for a given word w_i and the top N topic word for a given topic t,

$$CP(w_i) = \sum_{j}^{N-1} \frac{D(w_i, w_j)}{D(w_j)}$$

$$PMI(w_i) = \sum_{j}^{N-1} \frac{D(w_i, w_j)}{D(w_j) * D(w_i)}$$

$$NPMI(w_i) = \sum_{j}^{N-1} \frac{PMI(w_i)}{-log(D(w_j, w_i))}$$

where $D(W)$ is the number of document containing the set of words W.

[1] `https://www.reddit.com/r/SampleSize/`.

Level	Random	Closest	Furthest	Humans
1	0.986	0.972	1	0.793
2	0.983	0.978	0.996	0.513
3	0.968	0.959	0.970	0.064

Table 8.1: Results of the automated Word Intrusion task for each level comparing the three word intruder sampling techniques.

To better understand the performance of the automated version of the task, we compared three different sampling strategies for selecting the intruder word: 1) sampling from a random sibling topic, 2) sampling from the closest sibling topic, and 3) sampling from the furthest sibling topic. Distance between the siblings is computed as the JS divergence between their topic-word distributions. For the human-annotated task, we only used the random sampling strategy.

8.4 Results and Discussion

8.4.1 Results of the Word Intrusion task

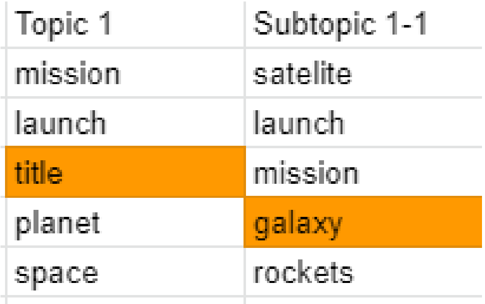
The results of the human-annotated version of the Word Intrusion task showed an average accuracy of 45.66% across all topics. However, this accuracy varied significantly with depth, as shown in Table 8.1. For level 1 topics, the performance was on par with the results reported in the original Word Intrusion task paper [Chang et al., 2009]. However, accuracy dropped dramatically for deeper levels, indicating that annotators had more difficulty finding the intruder word in these topics.

In contrast, the automated version of the task achieved extremely high accuracy, as shown in Table 8.1. The original study did not report the actual accuracy of the automated version, but only the Pearson correlation with human annotators [Lau et al., 2014]. Our results show that automating the task leads to much higher accuracy than the human-annotated version, with only slight variations based on depth and sampling method. Sampling the intruder from the furthest sibling led to slightly higher accuracy than sampling from the closest sibling, but the performance remained much higher than the human-annotated task in all cases.

8.4.2 The limitations of the Word Intrusion task in a hierarchical setting

Limitations of the human-annotated version

The results of our experiments reveal several limitations of the Word Intrusion task for evaluating the quality of hierarchical topic models.



Topic 1	Subtopic 1-1
mission	satelite
launch	launch
title	mission
planet	galaxy
space	rockets

Figure 8.2: Example of topics with an intruder. On the left, we have the parent topic and one of its children on the right. The intruder is highlighted in orange. Galaxy is a less obvious intruder than the title.

First, the task is difficult for human annotators, particularly for topics that are more esoteric and thus require domain knowledge. When running our experiments, one annotator actually remarked on the difficulty of not recognizing some technical words which hindered their ability to answer the survey. Another study made a similar observation when applying LDA to medical text [Arnold et al., 2016]. This situation is worsened for deeper sub-topics which by definition become more precise and require even more domain knowledge to understand. For example, the topic "covid-19" might contain the sub-topic "covid vaccines" which in turn might contains the topic "mRNA-based vaccines".

Second, as we delve deeper into the topic tree, sibling topics become increasingly thematically similar, as shown in Figure 8.1. Quantitative analysis revealed significant variations in the average JS divergence between topics at different depths (depth 1: 3500; depth 2: 1200; depth 3: 200), with deeper topics having smaller values. This indicates that deeper topics are less distinct from one another. Consequently, as we go deeper into the tree, the intruder word chosen for the word intrusion task becomes increasingly similar to the other words in the target topic (see Figure 8.2), making it more difficult for annotators to identify the intruder without specialized domain knowledge.

Hence, the human-annotated word intrusion task is unreliable if the annotators come from the general population. The alternative would be to consult domain experts but this approach would be either costly and lead to a smaller pool of annotators making the results less reliable once again. Moreover, as extracted topics can span a wide range of themes, we would need domain experts for each domain that the corpus presents. Hence, relying on domain experts may become prohibitively expensive if the corpus covers many different domains.

Third, because of the tree structure, the number of subtopics is exponentially greater than high-level topics. Hence, the high number of sub-topics in hierarchical topic models makes the task unscalable for human annotation, as it would be impractical to create a survey that covers a significant portion of the topics.

Limitations of the automated version

The automated version of the Word Intrusion task eliminates the issues related to human annotators (i.e. the lack of domain knowledge and the high number of topics to evaluate), but it has its own set of limitations.

First, the high accuracy of the automated version suggests that the task is too easy to solve for computers. Indeed, the accuracy of human annotators drops dramatically with topic depth as the topic becomes more esoteric and less distinct. However, the automated task is only barely affected by topic depth. Moreover, even though sampling from the closest sibling makes the task more difficult, the results remain above 95% accuracy. Finally, the automated version relies on simple features such as PMI and CP1. This indicates that for hierarchical topic models, the automated version of the task may not adequately capture the complexity of human judgment.

Second, we already achieve nearly 100% accuracy with the automated intrusion task. Hence, if we believe in the accuracy presented by this method, there isn't a lot of room for improvement, and the model performs extremely well. This would be wonderful if it was true. However, the low accuracy reported in the human-annotated task indicates that this is unlikely. Hence, the accuracy reported by the word intrusion task does not reflect the actual performance of the model.

8.4.3 Future directions for topic model evaluation

The limitations of current evaluation methods highlight the need for more reliable approaches that can better align with human judgment and accurately assess hierarchical topic models. We propose that evaluating topics and topic models should involve a multifaceted approach. While current methods such as perplexity, coherence, and the word intrusion task may be useful, they are not comprehensive and should eventually be replaced. To effectively evaluate topic models, we suggest considering the following dimensions:

- Coherence: Assesses the coherence of the set of words extracted by the model.
- Interpretability: Evaluates how easily human annotators can agree on a label for a topic
- Distinguishability: Assesses the distinctness of any two topics.
- Stability: Measures the variability in the top words of a model when run multiple times on the same corpus [Agrawal et al., 2018].
- Completeness: Evaluates the model's ability to extract all of the topics present in a corpus

Overall, these dimensions provide a useful framework for evaluating the quality of a topic model and can help to guide future research in this area.

8.5 Conclusion

In this study, we investigated the effectiveness of the Word Intrusion task for evaluating the quality of hierarchical topic models. Our results showed that the task is unreliable in this context for several reasons.

First, the human-annotated version of the task is extremely challenging for annotators to perform when evaluating deep sub-topics, which are often esoteric and difficult to understand. This makes it difficult for annotators to identify the intruder word, leading to low accuracy. Additionally, the task is not scalable for large topic trees, as there may be thousands of topics to evaluate.

Second, the automated version of the task achieves near-perfect accuracy when applied to deep sub-topics. However, this high accuracy does not necessarily reflect the quality of the topic model as judged by humans, as it is too easy for computers to learn. This suggests that the automated version of the task is not correlated with human judgment in the context of hierarchical topic models.

Our findings highlight the limitations of the Word Intrusion task for evaluating hierarchical topic models and underscore the need for more reliable evaluation methods. In particular, existing methods have not adequately considered the completeness of the extracted topics, which is an important aspect of topic model quality.

Therefore, this chapter demonstrated the need for a new approach to hierarchical topic model evaluation. Hence, the final chapter of this thesis will be focused on proposing a new solution for this challenging problem.

This chapter was submitted to ACL 2023.

9 Evaluating Hierarchical Topic Models Using a Labeled Dataset

9.1 Preamble

In this Section, we present our fifth article about using labeled data to perform a more holistic evaluation of topic models. This paper addresses the limitations of previous evaluation methods by focusing on the comprehensiveness and quality of the extracted topics, as well as the coherence of the generated taxonomy in the context of hierarchical topic models. By leveraging a well-known labeled dataset, Reuters-21578, our approach provides a more in-depth understanding of the model's performance and its ability to uncover expected topics, as well as its effectiveness in capturing smaller and unexpected topics.

Our experiments involved training 60 different models, including both hierarchical and flat models, with varying hyperparameters to quantitatively assess their ability to produce high-quality topics. The results demonstrate that our proposed label accuracy metric offers a more conservative measure of topic quality compared to coherence. Specifically, we demonstrate that while a low coherence score can be an indicator of poor topic quality, a high coherence score does not always indicate good topic quality.

Furthermore, we found that hierarchical topic models effectively extract small sub-topics, even those accounting for less than 1% of the data, with label accuracy as high as 37.9% for these small topics despite the existence of 90 labels. In contrast, larger topics achieved over 70% accuracy. This research showcases the value of using labeled data to evaluate topic models more comprehensively and highlights the strengths of hierarchical topic models in extracting meaningful insights from a diverse range of topics.

9.2 Introduction

Evaluating the quality of the extracted topics is crucial to ascertain their real-world utility. However, as these methods extract knowledge in an unsupervised manner, previous studies on topic model evaluation have been limited to evaluating the quality of the resulting topics. Hence, many methods have been proposed to study the performance of these models such as the perplexity and coherence measures [Newman et al., 2010, Doogan and Buntine, 2021, Bhatia et al., 2017].

Nonetheless, these measures have proved to be unrelated to human judgement [Chang et al., 2009, Doogan and Buntine, 2021, Bhatia et al., 2017], indicating that humans do not agree with these measures when it comes to the quality of the topics extracted. Recently, the word intrusion task has been proposed to evaluate the extracted topic quality [Chang et al., 2009]. While its initial implementation relies on human annotators, it can be automated without losing the link to human judgement [Lau et al., 2014].

However, all of the methods previously presented have failed to ask other essential questions about the extracted topics and the completeness of the results. For example: Do we extract every topic? How well do we extract them? Do we extract unexpected topics? And in the context of hierarchical topic models, is the hierarchy produced coherently?

Hence, in this article, we propose a method for evaluating topic models using a well-known labelled dataset (Reuters-21578 [Tekn, 2020]) but the method can be extended to other datasets. Our approach differs from previous methods by focusing on known topics that we expect to extract and their quality, providing a better understanding of the completeness of the model. Using known labels we can automatically name extracted topics. Afterwards, we can study whether the document topic distribution can predict the actual labels of the documents. We call this *label accuracy* and it provides a quantitative assessment of how well we fit the training set. Moreover, if more topics are extracted than expected we can study their relevance and unexpectedness. Finally, as the extracted topics exist in a hierarchy, we can analyze the coherence of the taxonomy produced from the known labels.

To perform our experiments, we trained 60 different models (30 hierarchical and 30 flat models) with various hyperparameters to understand how and if this new evaluation approach is able to help us determine quantitatively which model provides the best topics.

Results show that label accuracy provides a more conservative measure of topic quality compared to coherence. We show that while low coherence [Newman et al., 2010] is a good indicator of bad quality in topics, a high coherence score is not sufficient to determine the quality of a set of topics. We also compute the label accuracy for labels that account for less than 1% of the data and demonstrate that it is a good metric if we care about extracting small sub-topics. Precisely, we see that although we have 90 labels the accuracy of small topics can get as high as 37.9% while the largest topics achieve more than 70% accuracy. In that sense, we have noticed a logarithmic relationship between the number of documents per label and its accuracy as accuracy quickly goes up with the number of documents indicating that hierarchical topic models can extract small topics effectively.

9.3 Methodology

9.4 Overview

Our evaluation methodology consists of multiple steps. We aim to assess the stability and sensitivity of the topic models and compare the performance of hierarchical and flat models. To achieve this, we extract topics from our corpus using 60 variations of topic models (30 hierarchical and 30 flat models with different parameters as shown in Table 9.1) by training them on the Reuters dataset. The varying parameters include basic LDA parameters that control the topic-word and document-topic prior distributions, as well as the dynamic parameters controlling the creation of new topics during training.

Following this, we automatically assign labels to the topics by using the known labels from the corresponding dataset, based on the document-topic distribution. Next, for each document with n labels, we compare the top n+k labeled topics for that document to calculate label accuracy. Finally, we evaluate the results.

9.5 Corpus

For our experiments, we will employ the Reuters-21578 corpus[Tekn, 2020], a widely used dataset in the literature on topic models. Composed of English news articles primarily focused on business and politics, this corpus was used as it has detailed and multiple labels for each document.

We preprocessed the corpus by filtering relevant tokens using Spacy's Named Entity Recognition and Part-of-Speech tags and applied lemmatization. Consequently, our training set consists of 10788 documents, each labeled with one or more of the 90 tags in the corpus (e.g. wheat, gold, money-fx, etc.).

The label distribution is highly uneven, resembling a power-law distribution, with labels such as 'earn' or 'acq' constituting approximately 36% and 22% of the documents, respectively. In contrast, labels like 'rye' and 'castor-oil' appear only in a single document each.

9.6 Constructing and training the models

In our experiments, we utilized our HTMOT model presented in Chapter 6 stripped of its temporal component which is not evaluated here. Hence, mathematically we have the nHDP topic model albeit trained with our novel Gibbs sampling procedure instead of Stochastic Variational Inference (SVI).

As previously mentioned, Gibbs sampling performs better on small topics since it it asymptotically exact contrary to SVI [Blei et al., 2017]. Extracting these small topics is crucial since they may represent weak signals. Hence, we will also provide a quantitative analysis of the performance of labels spanning a small subset of the data.

We explored 60 distinct models, training 30 hierarchical models (nHDP) and 30 flat models (HDP) to highlight the importance of hierarchy for smaller labels. Each hierarchical/flat model pair shares the same set of parameters (refer to Table 9.1). For example, the set of parameters *S1* is used to train both a hierarchical and a flat model.

The parameters that we vary in each model are defined as follows: α: the rate at which we create new topics in the document trees. β: the rate at which we create new topics in the corpus tree. ϕ: the prior for the topic-word distribution. ϵ: the prior for the corpus and document-topic distributions.

Parameters	**alpha**	**beta**	**phi**	**epsilon**
S1	0.0001	0.02	0.1	0.5
S2	0.0001	0.02	0.1	0.5
S3	0.0001	0.02	0.1	0.5
S4	0.0001	0.02	0.1	0.5
S5	0.0001	0.02	0.1	0.5
S6	0.0001	0.02	0.1	0.5
A1	0.000005	0.02	0.1	0.5
A2	0.00001	0.02	0.1	0.5
A3	0.00005	0.02	0.1	0.5
A4	0.0005	0.02	0.1	0.5
A5	0.001	0.02	0.1	0.5
A6	0.005	0.02	0.1	0.5
B1	0.0001	0.001	0.1	0.5
B2	0.0001	0.002	0.1	0.5
B3	0.0001	0.004	0.1	0.5
B4	0.0001	0.1	0.1	0.5
B5	0.0001	0.2	0.1	0.5
B6	0.0001	0.4	0.1	0.5
E1	0.0001	0.02	0.1	0.001
E2	0.0001	0.02	0.1	0.01
E3	0.0001	0.02	0.1	0.02
E4	0.0001	0.02	0.1	0.1
E5	0.0001	0.02	0.1	2.
E6	0.0001	0.02	0.1	5.
P1	0.0001	0.02	0.001	0.5
P2	0.0001	0.02	0.01	0.5
P3	0.0001	0.02	0.02	0.5
P4	0.0001	0.02	0.5	0.5
P5	0.0001	0.02	1.	0.5
P6	0.0001	0.02	5.	0.5

Table 9.1: Parameters of the models trained

These 30 pairs of models are grouped as follows:

- 6 pairs of base models were trained with the same parameters to evaluate stability across multiple runs.
- 6 pairs of models with different values for alpha
- 6 pairs of models with different values for beta
- 6 pairs of models with different values for epsilon
- 6 pairs of models with different values for phi

9.7 Automatic titling

To automatically assign a label l to a topic we used a simple heuristic. For each model trained with the labelled corpus, we compute the label-topic distribution of label l by averaging the document-topic distribution of documents that have this label. If the model is hierarchical, this means we end up with a topic tree for label l with topic frequencies relative to this label, i.e. the proportion of topics for the subset of the data with label l.

Starting from the root, we select the topic with the highest frequency relative to that label. We do the same for the sub-topic of the selected topic until we reach a leaf. In the end, we have selected a branch of the tree containing the most frequent topic and sub-topic for the label l.

Next, we compare the known frequency of the label l with each topic of this branch and select the topic with the closest frequency i.e. the topic of the branch which optimizes $min_t(|frequency(t) - frequency(l)|)$. This topic will be given the label l.

For example (see Figure 9.1), if $l = gold$, the branch of the topic tree for that label that we will select could be interpreted as "Resources (45%) -> ore (20%) -> metal (10%)" because "Resources" had the highest frequency of topics at that level for the subset of documents with the label l and amongst subtopics "Resources", "ore" had the highest frequency for the subset of documents with the label l and so on. Then if l has a true frequency of 17%, we will label the topic "ore" with l since "ore" has the closest frequency at 15%.

This method is applied iteratively for each label. It is worth noting that a topic may have multiple labels in its title if it is selected by several labels.

This heuristic is simple by design and is an important hypothesis that may have a large impact on the performance of our evaluation methodology. Nonetheless, we will show that it is sufficient to provide interesting results.

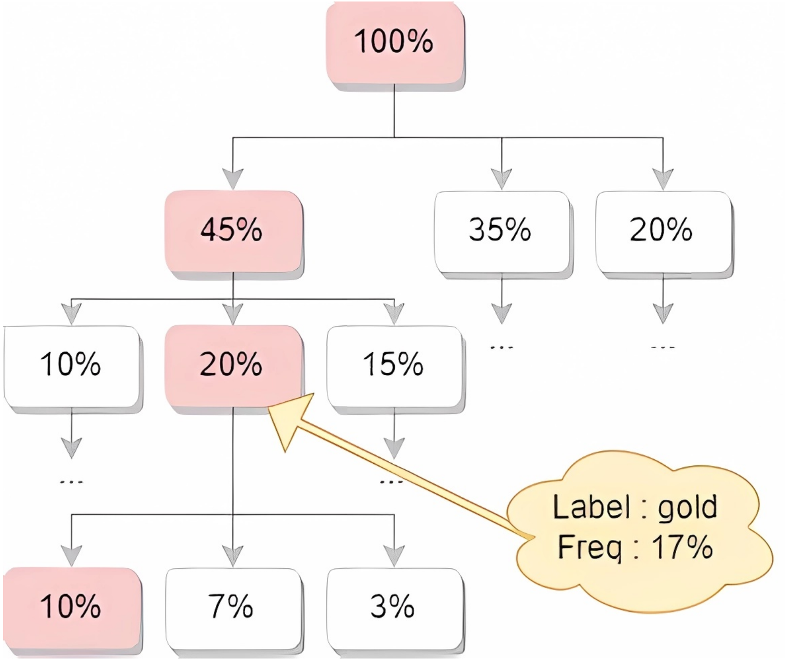


Figure 9.1: Example of topic tree for label l with the best branch highlighted in red.

9.8 Computing Top n+k label accuracy

To calculate the top n+k label accuracy, we order labeled topics by their document-topic distribution for each document by descending. Considering that document d has n labels, we choose the top n+k topics from the sorted list. We subsequently extract the labels given to these topics. Finally, using the set of extracted labels from the topics T and the known labels of document L, we determine the label accuracy for document d using the formula $\frac{|L \cap T|}{|L|}$. The overall top n+k label accuracy of the model is calculated as the average across all documents. The overall top n+k label accuracy of each label l is calculated as the average across all documents with that label l.

In addition to the overall top n+k label accuracy, we compute the small topic label accuracy, which excludes labels that correspond to more than 1% of the dataset. This exclusion accounts for 80% of the tags, or 72 tags in total.

This novel metric will be compared against the intrinsic UMass coherence measure. For a given topic denoted as t, alongside its list of words sorted by their associated probabilities $(w_0, \ldots, w_n)$, the UMass coherence is computed using the following formula: $\sum_{i<j} \log \frac{D(w_i,w_j)+1}{D(w_i)}$, where $D(W)$ represents the count of documents in the corpus that contain the set of words W, and $i < j$ indicates that the likelihood of occurrence is higher for word w_i compared to word w_j.

9.9 Results

In this Section, we will review the results of our experiments. We will start by comparing the coherence measure to the label accuracy measure. Next, we will compare the performance of the flat and hierarchical models. Then, we will study the hyperparameters' importance. Finally, we will study the stability of the topic models.

9.9.1 Coherence vs label accuracy

In Table 9.2, we display the metrics computed for all hierarchical models except those using Sx parameters as these are only used for stability analysis. Three were the worst in at least one metric and four were the best in at least one metric. The metrics are the average topic coherence and the top n+3 label accuracy which we will now refer to as top 3 label accuracy for simplicity's sake. The topic 3 label accuracy is computed for all the labels in each hierarchical model, in their flat counterpart (F), for small topics (S), and for both small topics in the flat model (F/S).

We observe that the model with the worst coherence (P1) did produce topics that are difficult to interpret. However, the model with the highest coherence (E1) is decisively not the best model. The label tree it produces is incoherent and most of the labels are pushed to the leaves of the tree. Consequently, this model has many topics which have been labelled multiple times indicating that the model could not separate the labels properly. Specifically, 81% of labels share a topic, and one topic shares as many as 34 labels. Moreover, this model created many duplicate topics (similar word distribution), with the majority of the topics being similar if not the same. Finally, we can observe that this model also has poor accuracy being the second worst.

The best-performing model is (B5) with the highest small label accuracy. Although its coherence is lower than (E1), its label tree is much more coherent and detailed. Most labels do not share co-labels meaning that the model is better at separating the labels into specific topics. For example, labels such as "gold" and "iron-steel" have been assigned their own topic contrary to the E1 model. Specifically, only 34% of labels share a topic, and one topic shares as many as 5 labels. B5 being the highest small topic accuracy, we also observed that small labeled topics are easily interpretable.

Hence, the coherence measure is good at determining if a set of topics are of bad quality. However, it is not sufficient in itself to determine if the topics are of good quality. A set of coherent but duplicate topics will yield a high coherence score even if this results in bad topic extraction overall. Moreover, high coherence does not guarantee that topics are well separated or that the inferred hierarchical structure of topics makes sense. Figure 9.2 shows that both label accuracy and coherence are not highly correlated which indicates they measure a different aspect of a model's performance.

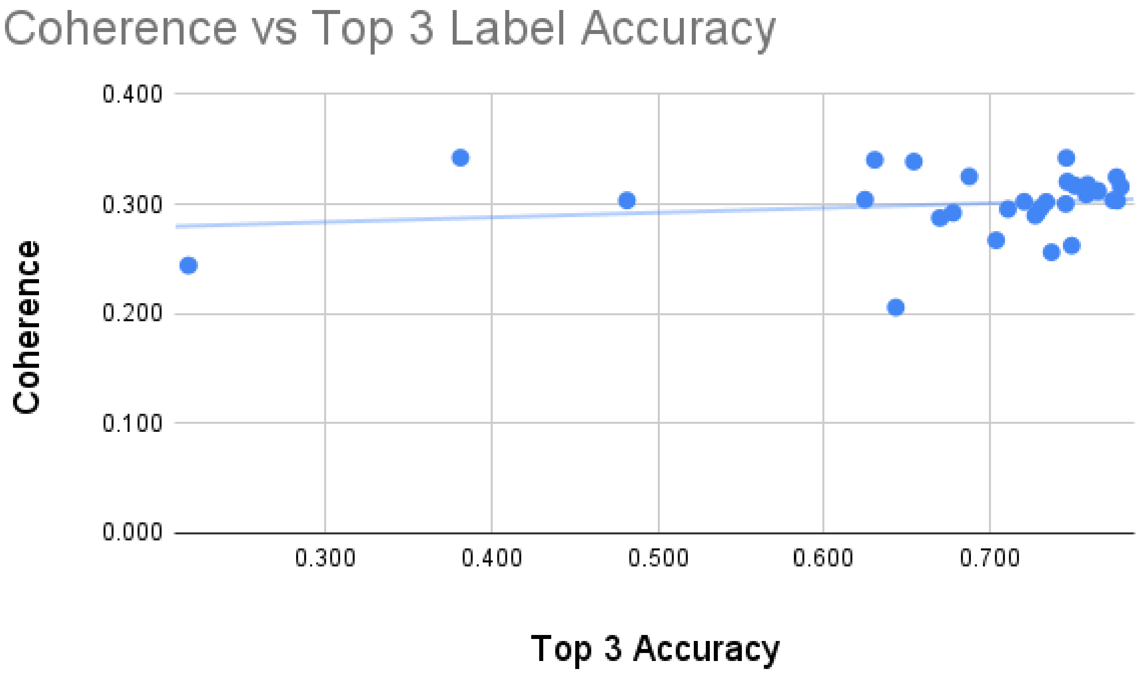


Figure 9.2: Coherence vs label accuracy across all models

Another way to ensure that the label accuracy represents the model's performance is to look at the discrepancy between the actual label size and the size of the topic with that label. In Table 9.3, we compare the worst and best models for small label accuracy. We see that for the best model, labels correspond to topics with a size that is closer to the actual label size.

Models	T3A	T3A (F)	T3A (S)	T3A (S/F)	Coh
P2	0.218	0.247	0.057	0.006	0.244
P1	0.643	0.543	0.178	0.004	0.206
A4	0.711	0.338	0.323	0.003	0.296
A2	0.778	0.590	0.271	0.012	0.316
E1	0.382	0.542	0.128	0.005	0.342
B5	0.727	0.350	0.379	0.006	0.290
E2	0.631	0.373	0.267	0.018	0.340

Table 9.2: Comparing best and worst models for each measure. T3A corresponds to the top 3 label accuracy. (F) corresponds to the equivalent flat model performance. (S) corresponds to the small topics' performance.

Tags	Real	B5	P2
nat-gas	proportion	0.89	16
gnp	1.19%	2.15	1.02
coffee	1.49%	1.41	12.09
trade	1.6%	2.25	1.06
crude	5.31%	3.13	6.62
money-fx	6.01%	1.53	6.49
acq	6.91%	20.57	8.6
MSE	24.56%	10.499	63.932

Table 9.3: Comparing the worst hierarchical topic model (P2) with the best small accuracy topic model on a set of random topics. We compare the real proportion of the tags in the data with the proportion of the topics with that label. We then compute the Mean Square Error (MSE) of this difference for both models.

We can also compare how the coherence and label accuracy metrics compare depending on the size of labels or topics. Since coherence is computed for each topic and label accuracy is computed for each label we cannot make a direct comparison. In the Figures 9.4 and 9.3, we plot these results and observe that there is a logarithmic relationship between label accuracy and size reminiscent of the formula $1 - e^{-x}$; indicating that the quality of topics quickly increases after only a small number of documents although it takes many documents to achieve near perfect accuracy. This implies that topic models could detect weak signals and emerging trends early as a few documents can produce relatively decent topics. However, for coherence, there is not such a clear relationship between topic size and coherence; the bigger topics do not seem to gain in coherence either. Nonetheless, a qualitative analysis of topics reveals that bigger topics are much easier to interpret.

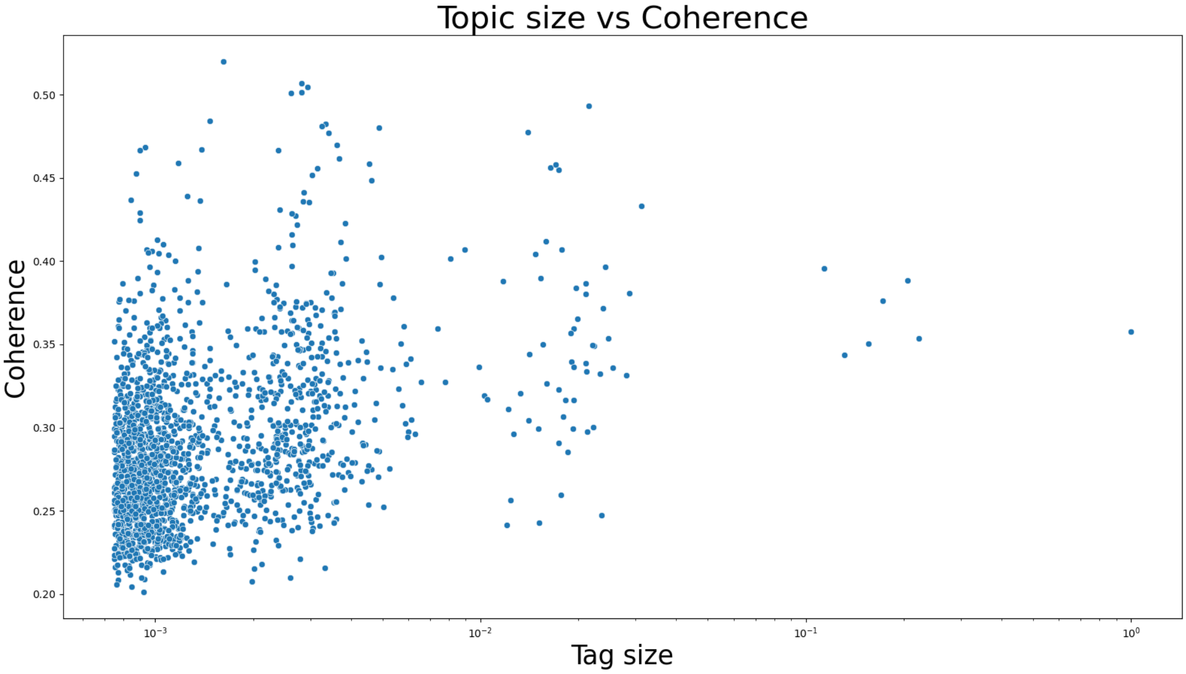


Figure 9.3: Topic coherence vs size. The x-axis uses a logarithmic scale.

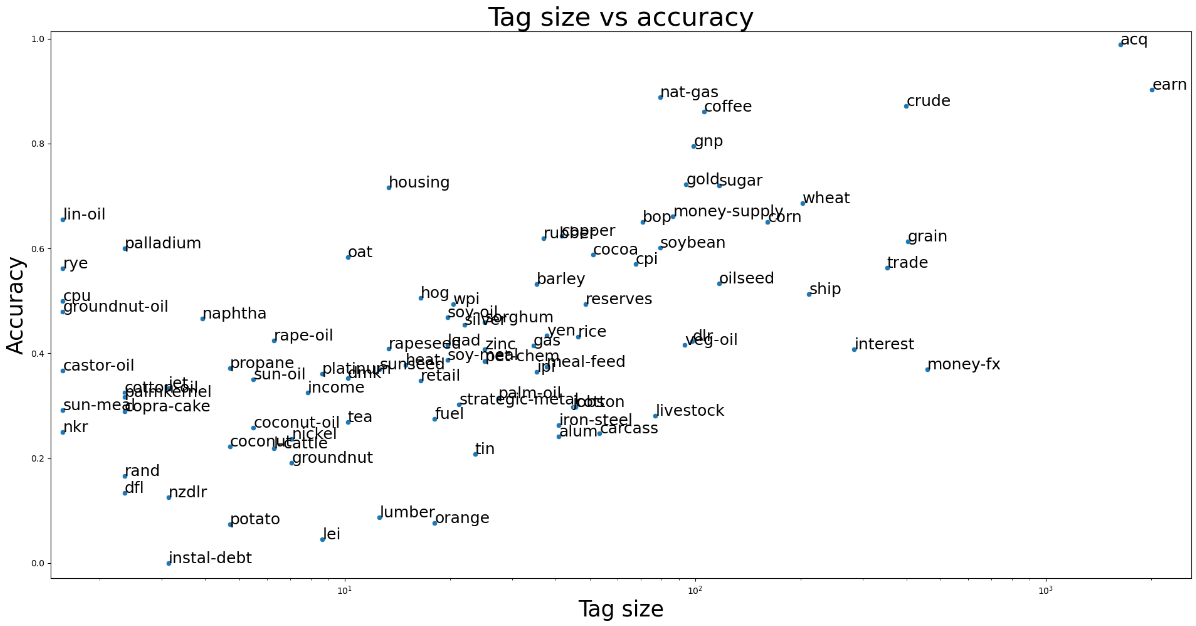


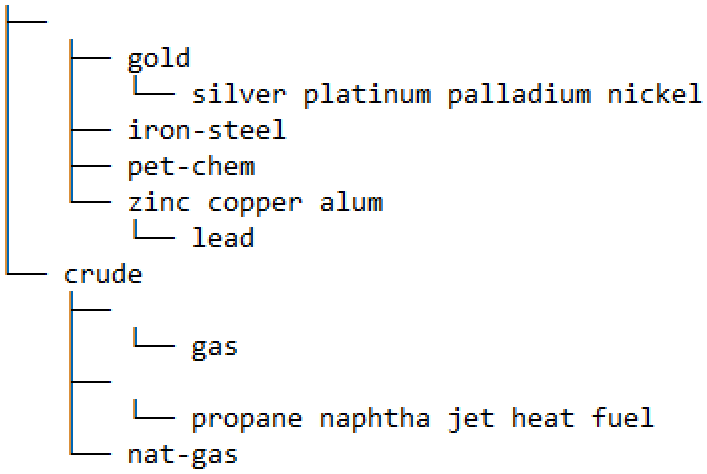
Figure 9.4: Label accuracy vs size. The x-axis uses a logarithmic scale.

Hence, we have demonstrated that while coherence is good at avoiding bad topics it is not sufficient to select good topic trees. The accuracy of small labels on the other hand provides us with a better understanding of the quality of a topic tree as a whole.

9.9.2 Flat vs hierarchical models

In Table 9.2, we can observe the label accuracy for the flat topic model for all the labels and the small one corresponding to a smaller set of documents. While the label accuracy can get close to 60%, it is mostly a reaction to the highly unbalanced labels in the corpus. Once, we focus on the smaller labels, this accuracy nearly drops to zero. This demonstrates the power of the hierarchical topic model to uncover smaller topics.

As we automatically label topics in a topic tree, we can also observe the coherence of the hierarchy produced. While the original labels are not structured in a hierarchy, we observe that the label taxonomy created from the topic tree is aligned with common sense knowledge (see Figure 9.5 for a sample). Thus, indicating that hierarchical topic models can produce coherent taxonomy from labeled documents.



```
├──
│   ├── gold
│   │   └── silver platinum palladium nickel
│   ├── iron-steel
│   ├── pet-chem
│   └── zinc copper alum
│       └── lead
└── crude
    ├──
    │   └── gas
    ├──
    │   └── propane naphtha jet heat fuel
    └── nat-gas
```

Figure 9.5: Selected sample of the label hierarchy produced. The entire label tree is too large to be shown entirely.

9.9.3 Hyper-parameter importance

Finally, we can study the hyper-parameter importance. We observe that ϵ and ϕ are positively correlated with label accuracy which controls document-topic and word-topic distributions, indicating that a more uniform distribution provides a better prior for this dataset. Nonetheless, for coherence higher values for ϕ and lower values for ϵ are preferable. For ϵ this discrepancy is interesting, although we have discussed that the model (E1) with the lowest value for ϵ is one of the worst models qualitatively and in terms of label accuracy.

If we believe in label accuracy, we may conclude that it is better to start with a uniform prior which does not setup the model in any specific local minima. Indeed, lower values of ϵ or ϕ will lead the model to select some random configuration for these distributions early on before it has been able to see the whole data; this is called the burn-in phase of the Gibbs procedure. On the other hand, starting with a uniform prior distribution forces the model to remain uniform until it has seen enough data that the empirical distribution in the data takes precedence over the prior. However, even higher values for these prior eventually lead to degrading performance since it will eventually have a higher weight than the data itself.

Considering the parameters which control the creation of topics during training. We see that higher β, which controls the rate at which we create new topics in the corpus tree, does not impact significantly label accuracy but does negatively impact coherence. We observe similar results for α: the rate at which we create new topics in the document trees. Expect that higher values for α is correlated with higher small label accuracy. Once again, these priors mostly impact the model during the burn-in phase of the Gibbs procedure.

T3A	T3A (S)	Coh
0.776	0.316	0.303
0.482	0.252	0.303
0.745	0.315	0.300
0.734	0.298	0.302
0.720	0.325	0.302
0.774	0.316	0.304

Table 9.4: Comparing the performance 5 models with the same parameters but a different random seed

9.9.4 Stability

In terms of stability, we observe that the model is mostly stable in terms of label accuracy with one model out of 5 that displays a significant deviation from the rest which we confirmed based on a qualitative analysis. However, coherence does not provide such observation. Once again, this demonstrates that our measure provides additional information about a model's performance.

9.9.5 Do we extract unexpected topics?

While quantitative analysis of topic models is important, it is necessary to remember that such models are not predictive. Hence, part of the reason we use topic models is to discover unexpected topics, meaning topics that do not corresponds to any known label. Indeed, it is important to note that while we have 90 labels in the dataset, we tend to extract about 1500 topics on average over all of the hierarchica models trained. Meaning that on average less than 5% of topics receive a label.

Hence, other unexpected topics have been extracted as well. We can look at the small unexpected topics extracted by the B5 model, these topics are displayed in Table 9.5. These topics are not specifically described by any of the labels present in the original dataset.

9.10 Conclusion

Our study introduces a novel method for evaluating hierarchical topic models based on labeled data. We trained hierarchical topic models on the Reuters-21578 dataset and used the known labels to evaluate the quality of the resulting topics. Our approach differs from previous methods by focusing on known topics that we expect to extract, providing a better understanding of the completeness of the model.

Ship attack .5%	Ore reserves 1.1%	Trade dispute .5%
iranian	estimate	semiconductor
attack	reserve	tariff
tanker	property	pact
missile	exploration	sanction
platform	total	impose
war	mining	market
oil	development	japanese
protect	prove	failure
ship	result	chip
shipping	program	computer

Table 9.5: A selection of small unexpected topics. These topics have a frequency of 0.49%, 1.11%, and 0.49% respectively.

We found that labels with a large number of documents yielded high accuracy above 70%, while smaller labels (1% of the data) had lower accuracy, but remained relatively high for a multi-class accuracy with 90 labels at 37.9%. Additionally, we observed a logarithmic relationship between label accuracy and size, indicating that even a small increase in the number of documents could greatly improve the quality of the extracted topics. This suggests that topic models can detect weak signals and emerging trends early, with just a few documents producing relatively decent topics.

Furthermore, we demonstrated that coherence alone is not sufficient to select a good topic tree, and the accuracy of small labels provides a better understanding of the quality of the topic tree. Our approach also allowed us to discover unexpected topics, such as trade disputes or ore reserves, that would have been missed by traditional evaluation methods. Lastly, we have shown that hierarchical topic models produce relatively coherent label taxonomy.

Future research could build on our approach by developing better evaluation methods that consider not only the quality of topics extracted but also the ability to extract expected topics. Another direction for future research is to measure the unexpectedness of extracted topics since topic models are often used to discover unknown patterns in the data.

This Chapter proposed a new solution for evaluating topic models, especially hierarchical ones. As the final Chapter of this thesis it demonstrates the capabilities of hierarchical topic models such as HTMOT to extract topics, especially small or unexpected ones.

This Chapter was submitted to RANLP 2023.

Part V

CONCLUSION, LIMITATIONS AND FUTURE WORK

In the methodology Section of this thesis, a comprehensive investigation was conducted into various aspects of natural language processing (NLP) tasks, including coreference resolution and topic modeling. State-of-the-art techniques were utilized, and the performance of different embeddings and models was compared, considering factors such as size, predictive performance, and training time. A hierarchical and temporal topic modeling approach was employed for trend analysis, providing advantages over existing methods by enabling fine-grained analysis, tracking, and discovery of topic interactions.

For coreference resolution (i.e. the research question Q1), it was found that a smaller model using only character embeddings achieved 86% of the performance of a larger model using multiple embeddings (Elmo, GloVe, and character embeddings), while being only 1.2% of its size. Surprisingly, the smallest model using solely character embeddings even outperformed the previous state-of-the-art word embedding method, Word2Vec. The study also revealed that adding additional embeddings did not significantly improve performance, suggesting diminishing returns in terms of predictive performance per embedding. Interestingly, the largest model converged faster during training, indicating a weak correlation between size and run-time.

In the context of topic modeling (i.e. the research question Q2), a new model capable of modeling hierarchy and time jointly was proposed, providing a more detailed understanding of a corpus by incorporating sub-topics that represent specific events. A novel implementation of Gibbs sampling for hierarchical topic models was developed for training, offering a more exact alternative to stochastic variational inference (SVI). Additionally, the study demonstrated how extracting entities enhanced the interpretation and understanding of topics at a deeper level.

Topic tracking was explored (i.e. the research question Q3), introducing a semantic-based method that performed comparably to a state-of-the-art lexical-based approach. This highlighted the value of incorporating semantic information in tracking topics. The challenges associated with tracking topics in a hierarchy, such as differentiating between topics and sub-topics, and evaluating the tracking quality of deeper, more esoteric topics, were discussed. It was suggested that future studies should explore hybrid methods that integrate both lexical and semantic information for topic tracking.

In the evaluation part of the thesis (i.e. the research question Q4), the effectiveness of various evaluation methods for hierarchical topic models was examined. In particular, we experimented with the Word Intrusion task for evaluating hierarchical topic models and identified several limitations. The human-annotated version of the task proved challenging for annotators, particularly when evaluating deep sub-topics, leading to low accuracy. The automated version achieved near-perfect accuracy but failed to align with human judgment, suggesting it was too easy for computers to learn. These findings emphasized the necessity for more reliable evaluation methods that consider the completeness of extracted topics.

A novel approach based on labeled data was proposed, demonstrating its potential in evaluating hierarchical topic models. The method focused on known topics and their extraction, providing insights into the completeness of the model. The results showed that labels associated with a large number of documents yielded higher accuracy in topic extraction. However, smaller labels still achieved relatively high accuracy for a multi-class setting. The relationship between label accuracy and size followed a logarithmic trend, suggesting that even a small increase in the number of documents significantly improved the quality of extracted topics. This indicated the ability of topic models to detect weak signals and emerging trends with just a few relevant documents.

The coherence metric was found to be insufficient to evaluate a model as a whole as high coherence could be achieved even when all the topics extracted were the same. On the other hand, label accuracy provided valuable insights into the topic model's performance. The approach also uncovered unexpected topics that would have been missed by traditional evaluation methods, showcasing the capabilities of hierarchical topic models in uncovering novel patterns and information.

Future research should focus on developing evaluation methods that consider both the quality of extracted topics and the ability to extract expected topics. Additionally, measuring the unexpectedness of extracted topics could provide further insights, as topic models are often employed to discover previously unknown patterns. Moreover, with the advent of Large Language Models, new techniques for topic modelling extraction and interpretation could be envisaged.

While the methodology shows promise in providing valuable insights for organizations by identifying emerging weak signals, there are certain limitations that need to be acknowledged. These include the difficulty of evaluating unsupervised models, the necessity of manual topic interpretation, and the lack of online learning capabilities, requiring re-running the entire methodology whenever the corpus grows.

Future work could address these limitations by focusing on developing better methods for evaluating topic models, improving automatic labeling of topics for easier interpretation based on large language models, and integrating sentiment analysis to understand feelings towards weak signals and emerging trends. Additionally, exploring the applicability of the methodology to various domains and creating more sophisticated techniques for topic tracking and interaction analysis could further enhance organizations' ability to make informed strategic decisions based on evolving landscapes in their industries.

References

Bibliography

[Agrawal et al., 2018] Agrawal, A., Fu, W., and Menzies, T. (2018). What is wrong with topic modeling? and how to fix it using search-based software engineering. *Information and Software Technology*, 98:74–88.

[akhavanhariri et al., 2022] akhavanhariri, e., Mansouri, A., and Najafabadi, H. (2022). Identification of hot topics and trends in knowledge and information science, based on text mining techniques. *Iranian Journal of Information Processing & Management*, 38.

[Alghamdi and Alfalqi, 2015] Alghamdi, R. and Alfalqi, K. (2015). A survey of topic modeling in text mining. *International Journal of Advanced Computer Science and Applications*, 6(1).

[Allan et al., 2003] Allan, J., Lavrenko, V., and Connell, M. E. (2003). A month to topic detection and tracking in hindi. *ACM Transactions on Asian Language Information Processing (TALIP)*, 2(2):85–100.

[Almeida and Xexéo, 2019] Almeida, F. and Xexéo, G. (2019). Word embeddings: A survey. *CoRR*, abs/1901.09069.

[AlSumait et al., 2008] AlSumait, L., Barbará, D., and Domeniconi, C. (2008). On-line lda: Adaptive topic models for mining text streams with applications to topic detection and tracking. In *2008 eighth IEEE international conference on data mining*, pages 3–12. IEEE.

[Andrus, 1997] Andrus, D. L. (1997). Book review: Inside the tornado: Marketing strategies from silicon valley's cutting edge. *Journal of Marketing*, 61(2).

[Ansoff, 1975] Ansoff, H. I. (1975). Managing strategic surprise by response to weak signals. *California management review.*

[Arnold et al., 2016] Arnold, C. W., Oh, A., Chen, S., and Speier, W. (2016). Evaluating topic model interpretability from a primary care physician perspective. *Computer methods and programs in biomedicine*, 124:67–75.

[Azzam et al., 1999] Azzam, S., Humphreys, K., and Gaizauskas, R. (1999). Using coreference chains for text summarization. In *Coreference and Its Applications*, CorefApp '99, page 77–84, USA. Association for Computational Linguistics.

[Barde and Bainwad, 2017] Barde, B. V. and Bainwad, A. M. (2017). An overview of topic modeling methods and tools. In *2017 International Conference on Intelligent Computing and Control Systems (ICICCS).* IEEE.

[Barhom et al., 2019] Barhom, S., Shwartz, V., Eirew, A., Bugert, M., Reimers, N., and Dagan, I. (2019). Revisiting joint modeling of cross-document entity and event coreference resolution. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4179–4189, Florence, Italy. Association for Computational Linguistics.

[Bastani et al., 2019] Bastani, K., Namavari, H., and Shaffer, J. (2019). Latent dirichlet allocation (lda) for topic modeling of the cfpb consumer complaints. *Expert Systems with Applications*, 127:256–271.

[Bejan and Harabagiu, 2010] Bejan, C. and Harabagiu, S. (2010). Unsupervised event coreference resolution with rich linguistic features. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1412–1422, Uppsala, Sweden. Association for Computational Linguistics.

[Berardi et al., 2015] Berardi, G., Esuli, A., and Marcheggiani, D. (2015). Word embeddings go to italy: A comparison of models and training datasets. In *IIR*.

[Bhatia et al., 2017] Bhatia, S., Lau, J. H., and Baldwin, T. (2017). An automatic approach for document-level topic model evaluation. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 206–215, Vancouver, Canada. Association for Computational Linguistics.

[Birjali et al., 2021] Birjali, M., Kasri, M., and Beni-Hssane, A. (2021). A comprehensive survey on sentiment analysis: Approaches, challenges and trends. *Knowledge-Based Systems*, 226:107134.

[Biswas, 2023] Biswas, S. S. (2023). Role of chat gpt in public health. *Annals of Biomedical Engineering*, pages 1–2.

[Blei et al., 2004] Blei, D. M., Griffiths, T. L., Jordan, M. I., and Tenenbaum, J. B. (2004). Hierarchical topic models and the nested chinese restaurant process. *Advances in neural information processing systems*, 16(16):17–24.

[Blei et al., 2017] Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. (2017). Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877.

[Blei and Lafferty, 2006] Blei, D. M. and Lafferty, J. D. (2006). Dynamic topic models. In *Proceedings of the 23rd international conference on Machine learning*, pages 113–120.

[Blei and Lafferty, 2007] Blei, D. M. and Lafferty, J. D. (2007). A correlated topic model of science. *The Annals of Applied Statistics*.

[Blei et al., 2003] Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3(null):993–1022.

[Bojanowski et al., 2017] Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

[Breitzman and Thomas, 2015] Breitzman, A. and Thomas, P. (2015). The emerging clusters model: A tool for identifying emerging technologies across multiple patent systems. *Research Policy*, 44(1).

[Cambria and White, 2014] Cambria, E. and White, B. (2014). Jumping nlp curves: A review of natural language processing research. *IEEE Computational intelligence magazine*, 9(2):48–57.

[Chang et al., 2009] Chang, J., Gerrish, S., Wang, C., Boyd-graber, J., and Blei, D. (2009). Reading tea leaves: How humans interpret topic models. In Bengio, Y., Schuurmans, D., Lafferty, J., Williams, C., and Culotta, A., editors, *Advances in Neural Information Processing Systems*, volume 22, pages 288–296. Curran Associates, Inc.

[Choubey and Huang, 2017] Choubey, P. K. and Huang, R. (2017). Event coreference resolution by iteratively unfolding inter-dependencies among events. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2124–2133, Copenhagen, Denmark. Association for Computational Linguistics.

[Coffman, 1997] Coffman, B. (1997). Weak signal research, part i-v. *Journal of Transition Management.*

[Cybulska and Vossen, 2014] Cybulska, A. and Vossen, P. (2014). Using a sledgehammer to crack a nut? lexical diversity and event coreference resolution. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 4545–4552, Reykjavik, Iceland. European Language Resources Association (ELRA).

[Dagan et al., 1997] Dagan, I., Lee, L., and Pereira, F. (1997). Similarity-based methods for word sense disambiguation. In *35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics*, pages 56–63, Madrid, Spain. Association for Computational Linguistics.

[Devlin et al., 2019] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

[Doogan and Buntine, 2021] Doogan, C. and Buntine, W. (2021). Topic model or topic twaddle? re-evaluating semantic interpretability measures. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3824–3848, Online. Association for Computational Linguistics.

[dos Santos and Guimarães, 2015] dos Santos, C. and Guimarães, V. (2015). Boosting named entity recognition with neural character embeddings. In *Proceedings of the Fifth Named Entity Workshop*, volume abs/1505.05008, pages 25–33, Beijing, China. Association for Computational Linguistics.

[Dos Santos and Zadrozny, 2014] Dos Santos, C. N. and Zadrozny, B. (2014). Learning character-level representations for part-of-speech tagging. In *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*, volume 32 of *ICML'14*, page II–1818–II–1826. JMLR.org.

[El Akrouchi et al., 2021] El Akrouchi, M., Benbrahim, H., and Kassou, I. (2021). End-to-end lda-based automatic weak signal detection in web news. *Knowledge-Based Systems*, 212:106650.

[Fan et al., 2021] Fan, W., Guo, Z., Bouguila, N., and Hou, W. (2021). Clustering-based online news topic detection and tracking through hierarchical bayesian nonparametric models. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2126–2130.

[Glenn and Gordon, 2009] Glenn, J. and Gordon, T. (2009). Environmental scanning. *Futures research methodology.*

[Gregoriades et al., 2021] Gregoriades, A., Pampaka, M., Herodotou, H., and Christodoulou, E. (2021). Supporting digital content marketing and messaging through topic modelling and decision trees. *Expert Systems with Applications*, 184:115546.

[Gromann and Declerck, 2018] Gromann, D. and Declerck, T. (2018). Comparing pretrained multilingual word embeddings on an ontology alignment task. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

[Held et al., 2021] Held, W., Iter, D., and Jurafsky, D. (2021). Focus on what matters: Applying discourse coherence theory to cross document coreference. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1406–1417, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

[Hiltunen, 2007a] Hiltunen, E. (2007a). Where do future oriented people find weak signals. *FFRC eBook*.

[Hiltunen, 2007b] Hiltunen, E. (2007b). Where do future-oriented people find weak signals. *FFRC eBook*, 2:2007.

[Hiltunen, 2008] Hiltunen, E. (2008). The future sign and its three dimensions. *Futures*, 40:247–260.

[Hong et al., 2011] Hong, L., Yin, D., Guo, J., and Davison, B. D. (2011). Tracking trends: Incorporating term volume into temporal topic models. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, page 484–492, New York, NY, USA. Association for Computing Machinery.

[Hoyle et al., 2021] Hoyle, A. M., Goel, P., Peskov, D., Hian-Cheong, A., Boyd-Graber, J. L., and Resnik, P. (2021). Is automated topic model evaluation broken?: The incoherence of coherence. *CoRR*, abs/2107.02173.

[Humphreys et al., 1997] Humphreys, K., Gaizauskas, R., and Azzam, S. (1997). Event coreference for information extraction. In *Operational Factors in Practical, Robust Anaphora Resolution for Unrestricted Texts*.

[Joshi et al., 2019] Joshi, M., Levy, O., Zettlemoyer, L., and Weld, D. (2019). BERT for coreference resolution: Baselines and analysis. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5803–5808, Hong Kong, China. Association for Computational Linguistics.

[Kenyon-Dean et al., 2018] Kenyon-Dean, K., Cheung, J. C. K., and Precup, D. (2018). Resolving event coreference with supervised representation learning and clustering-oriented regularization. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 1–10, New Orleans, Louisiana. Association for Computational Linguistics.

[Kim and Lee, 2017] Kim, J. and Lee, C. (2017). Novelty-focused weak signal detection in futuristic data: Assessing the rarity and paradigm unrelatedness of signals. *Technological Forecasting and Social Change*, 120.

[Kim et al., 2016] Kim, J., Park, Y., and Lee, Y. (2016). A visual scanning of potential disruptive signals for technology roadmapping: investigating keyword cluster, intensity, and relationship in futuristic data. *Technology Analysis & Strategic Management*, 28(10).

[Kim et al., 2013] Kim, S., Kim, Y.-E., Bae, K.-J., Choi, S.-B., Park, J.-K., Koo, Y.-D., Park, Y.-W., Choi, H.-K., Kang, H.-M., and Hong, S.-W. (2013). Nest: A quantitative model for detecting emerging trends using a global monitoring expert network and bayesian network. *Futures*, 52:59–73.

[Kim et al., 2020] Kim, S., Park, H., and Lee, J. (2020). Word2vec-based latent semantic analysis (w2v-lsa) for topic modeling: A study on blockchain technology trend analysis. *Expert Systems with Applications*, 152:113401.

[Korfiatis et al., 2019] Korfiatis, N., Stamolampros, P., Kourouthanassis, P., and Sagiadinos, V. (2019). Measuring service quality from unstructured data: A topic modeling application on airline passengers' online reviews. *Expert Systems with Applications*, 116:472–486.

[Kuusi and Hiltunen, 2012] Kuusi, O. and Hiltunen, E. (2012). The signification process of the future sign. *Journal of Futures Studies*, 16.

[Lau et al., 2014] Lau, J. H., Newman, D., and Baldwin, T. (2014). Machine reading tea leaves: Automatically evaluating topic coherence and topic model quality. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 530–539, Gothenburg, Sweden. Association for Computational Linguistics.

[Lee et al., 2012] Lee, H., Recasens, M., Chang, A., Surdeanu, M., and Jurafsky, D. (2012). Joint entity and event coreference resolution across documents. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 489–500, Jeju Island, Korea. Association for Computational Linguistics.

[Lee et al., 2017] Lee, K., He, L., Lewis, M., and Zettlemoyer, L. (2017). End-to-end neural coreference resolution. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 188–197, Copenhagen, Denmark. Association for Computational Linguistics.

[Lee and Park, 2018] Lee, Y.-J. and Park, J.-Y. (2018). Identification of future signal based on the quantitative and qualitative text mining: a case study on ethical issues in artificial intelligence. *Quality & Quantity: International Journal of Methodology*, 52(2).

[Li et al., 2018] Li, H., Li, X., Caragea, D., and Caragea, C. (2018). Comparison of word embeddings and sentence encodings as generalized representations for crisis tweet classification tasks. *Proceedings of ISCRAM Asia Pacific.*

[Li et al., 2020] Li, Z., Wallace, E., Shen, S., Lin, K., Keutzer, K., Klein, D., and Gonzalez, J. E. (2020). Train large, then compress: Rethinking model size for efficient training and inference of transformers.

[Liu et al., 2020a] Liu, H., Chen, Z., Tang, J., Zhou, Y., and Liu, S. (2020a). Mapping the technology evolution path: a novel model for dynamic topic detection and tracking. *Scientometrics*, 125(3):2043–2090.

[Liu et al., 2018] Liu, H., Ma, M., Huang, L., Xiong, H., and He, Z. (2018). Robust neural machine translation with joint textual and phonetic embedding. *arXiv preprint arXiv:1810.06729.*

[Liu et al., 2020b] Liu, Q., Kusner, M. J., and Blunsom, P. (2020b). A survey on contextual embeddings.

[Lu and Ng, 2018] Lu, J. and Ng, V. (2018). Event coreference resolution: A survey of two decades of research. In Lang, J., editor, *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, pages 5479–5486. ijcai.org.

[Martin and Johnson, 2015] Martin, F. and Johnson, M. (2015). More efficient topic modelling through a noun only approach. In *Proceedings of the Australasian Language Technology Association Workshop 2015*, pages 111–115, Parramatta, Australia.

[Miculicich Werlen and Popescu-Belis, 2017] Miculicich Werlen, L. and Popescu-Belis, A. (2017). Using coreference links to improve Spanish-to-English machine translation. In *Proceedings of the 2nd Workshop on Coreference Resolution Beyond OntoNotes (CORBON 2017)*, pages 30–40, Valencia, Spain. Association for Computational Linguistics.

[Mikolov et al., 2013] Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space.

[Mimno et al., 2007] Mimno, D., Li, W., and McCallum, A. (2007). Mixtures of hierarchical topics with pachinko allocation. In *Proceedings of the 24th international conference on Machine learning*, pages 633–640.

[Moosavi and Strube, 2016] Moosavi, N. S. and Strube, M. (2016). Which coreference evaluation metric do you trust? a proposal for a link-based entity aware metric. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 632–642, Berlin, Germany. Association for Computational Linguistics.

[Mühlroth and Grottke, 2018] Mühlroth, C. and Grottke, M. (2018). A systematic literature review of mining weak signals and trends for corporate foresight. *Journal of Business Economics*, 88(5):643–687.

[Nallapati et al., 2007] Nallapati, R. M., Ditmore, S., Lafferty, J. D., and Ung, K. (2007). Multiscale topic tomography. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 520–529.

[Newman et al., 2010] Newman, D., Lau, J. H., Grieser, K., and Baldwin, T. (2010). Automatic evaluation of topic coherence. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, page 100–108, USA. Association for Computational Linguistics.

[Nickel and Kiela, 2017] Nickel, M. and Kiela, D. (2017). Poincaré embeddings for learning hierarchical representations. *Advances in neural information processing systems*, 30.

[Paisley et al., 2015] Paisley, J., Wang, C., Blei, D. M., and Jordan, M. I. (2015). Nested hierarchical dirichlet processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(2):256–270.

[Park and Cho, 2017] Park, C. and Cho, S. (2017). Future sign detection in smart grids through text mining. *Energy Procedia*, 128. International Scientific Conference "Environmental and Climate Technologies", CONECT 2017, 10-12 May 2017, Riga, Latvia.

[Patel and Bhattacharyya, 2017] Patel, K. and Bhattacharyya, P. (2017). Towards lower bounds on number of dimensions for word embeddings. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 31–36, Taipei, Taiwan. Asian Federation of Natural Language Processing.

[Peirce, 1868] Peirce, C. S. (1868). Some consequences of four incapacities. *Journal of Speculative Philosophy.*

[Peloso, 2020] Peloso, A. (2020). Leveraging the power of micro-macro trends in contemporary organizations. *Journal of Applied Business and Economics*, 22(4).

[Peng et al., 2016] Peng, H., Song, Y., and Roth, D. (2016). Event detection and co-reference with minimal supervision. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 392–402, Austin, Texas. Association for Computational Linguistics.

[Pennington et al., 2014] Pennington, J., Socher, R., and Manning, C. (2014). GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

[Peters et al., 2018] Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

[Pradhan et al., 2012] Pradhan, S., Moschitti, A., Xue, N., Uryupina, O., and Zhang, Y. (2012). CoNLL-2012 shared task: Modeling multilingual unrestricted coreference in OntoNotes. In *Joint Conference on EMNLP and CoNLL - Shared Task*, pages 1–40, Jeju Island, Korea. Association for Computational Linguistics.

[Pujara and Skomoroch, 2012] Pujara, J. and Skomoroch, P. (2012). Large-scale hierarchical topic models. In *NIPS Workshop on Big Learning*, volume 128.

[Song et al., 2008] Song, Y., Zhang, L., and Giles, C. L. (2008). A non-parametric approach to pair-wise dynamic topic correlation detection. In *2008 Eighth IEEE International Conference on Data Mining*, pages 1031–1036. IEEE.

[Stahlberg, 2020] Stahlberg, F. (2020). Neural machine translation: A review. *Journal of Artificial Intelligence Research*, 69:343–418.

[Stylianou and Vlahavas, 2021] Stylianou, N. and Vlahavas, I. (2021). A neural entity coreference resolution review. *Expert Systems with Applications*, 168:114466.

[Su et al., 2008] Su, J., Yang, X., Hong, H., Tateisi, Y., and Tsujii, J. (2008). Coreference Resolution in Biomedical Texts: a Machine Learning Approach. In Ashburner, M., Leser, U., and Rebholz-Schuhmann, D., editors, *Ontologies and Text Mining for Life Sciences : Current Status and Future Perspectives*, number 08131 in Dagstuhl Seminar Proceedings, Dagstuhl, Germany. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, Germany.

[Sukthanker et al., 2018] Sukthanker, R., Poria, S., Cambria, E., and Thirunavukarasu, R. (2018). Anaphora and coreference resolution: A review.

[Teh et al., 2006] Teh, Y. W., Jordan, M. I., Beal, M. J., and Blei, D. M. (2006). Hierarchical dirichlet processes. *Journal of the American Statistical Association*, 101(476):1566–1581.

[Tekn, 2020] Tekn, Y. (2020). Optimization of lda parameters. In *2020 28th Signal Processing and Communications Applications Conference (SIU)*, pages 1–4.

[Tschannen et al., 2018] Tschannen, M., Bachem, O., and Lucic, M. (2018). Recent advances in autoencoder-based representation learning. *arXiv preprint arXiv:1812.05069.*

[Wang et al., 2019] Wang, H.-C., Hsu, T.-T., and Sari, Y. (2019). Personal research idea recommendation using research trends and a hierarchical topic model. *Scientometrics*, 121(3):1385–1406.

[Wang and McCallum, 2006] Wang, X. and McCallum, A. (2006). Topics over time: A non-markov continuous-time model of topical trends. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '06, page 424–433, New York, NY, USA. Association for Computing Machinery.

[Wiseman et al., 2016] Wiseman, S., Rush, A. M., and Shieber, S. M. (2016). Learning global features for coreference resolution. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 994–1004, San Diego, California. Association for Computational Linguistics.

[Xiao and Stibor, 2010] Xiao, H. and Stibor, T. (2010). Efficient collapsed gibbs sampling for latent dirichlet allocation. In Sugiyama, M. and Yang, Q., editors, *Proceedings of 2nd Asian Conference on Machine Learning*, volume 13 of *Proceedings of Machine Learning Research*, pages 63–78, Tokyo, Japan. JMLR Workshop and Conference Proceedings.

[Xu et al., 2019] Xu, G., Meng, Y., Chen, Z., Qiu, X., Wang, C., and Yao, H. (2019). Research on topic detection and tracking for online news texts. *IEEE access*, 7:58407–58418.

[Yang et al., 2015] Yang, G., Wen, D., Kinshuk, Chen, N.-S., and Sutinen, E. (2015). A novel contextual topic model for multi-document summarization. *Expert Systems with Applications*, 42(3):1340–1352.

[Yoon, 2012] Yoon, J. (2012). Detecting weak signals for long-term business opportunities using text mining of web news. *Expert Systems with Applications*, 39.

[Zhao et al., 2023] Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., Dong, Z., et al. (2023). A survey of large language models. *arXiv preprint arXiv:2303.18223*.

[Zhe et al., 2011] Zhe, G., Zhe, J., Shoushan, L., Bin, T., Xinxin, N., and Yang, X. (2011). An adaptive topic tracking approach based on single-pass clustering with sliding time window. In *Proceedings of 2011 International Conference on Computer Science and Network Technology*, volume 2, pages 1311–1314. IEEE.

[Zhou and Chen, 2013] Zhou, X. and Chen, L. (2013). Event detection over twitter social media streams. *The VLDB Journal*, 23(3):381–400.

[Zhu et al., 2016] Zhu, M., Zhang, X., and Wang, H. (2016). A lda based model for topic evolution: Evidence from information science journals. In *Proceedings of the 2016 International Conference on Modeling, Simulation and Optimization Technologies and Applications (MSOTA 2016)*, pages 49–54.

[Zhu et al., 2017] Zhu, X., Klabjan, D., and Bless, P. N. (2017). Unsupervised terminological ontology learning based on hierarchical topic modeling. In *2017 IEEE International Conference on Information Reuse and Integration (IRI)*, pages 32–41.