Utrecht University

UTRECHT UNIVERSITY

Department of Information and Computing Science

Artificial Intelligence Master Thesis

Exploring the Thematic Coherence of Fake and LLM-generated news: A Topic Modeling Approach

Supervisor:
Dr. D. Dell'Anna

Second examiner:
Dr. D.P. Nguyen

Author:
R.C.D. Willems
6505546

January 20, 2025

Abstract

Coherence is a fundamental property of well-written texts. As fake news continues to proliferate online and the prevalence of LLM-generated texts grows, the trustworthiness of news articles is broadly put under pressure. Therefore, robust tools to analyze and help understand the language used in such texts are increasingly important. This research explores the thematic coherence as of such texts, which pertains to a the ability of a text to stay focused on its core theme(s). Existing coherence models largely focus on local coherence or use opaque global coherence models and their applications in determining the veracity and origin of news articles is limited. In this study, a novel, interpretable method for modeling thematic coherence is developed as first research contribution. It leverages topic modeling and divergence metrics to assess the alignment of the themes discussed in news articles. The method is evaluated on the traditional sentence ordering evaluation task, which highlights the limited effectiveness of that task on capturing thematic coherence. To overcome that, a new evaluation task is proposed as second research contribution and the findings demonstrate that the proposed method can effectively distinguish thematically coherent from incoherent articles. When applied to detecting human-written fake news, the method shows significant differences between the thematic coherence of real and fake news articles and yields modest performance in standalone classification. For LLM-generated news articles, the method reveals slight thematic differences compared to human-written articles, with limited effectiveness in distinguishing between the two. The method gives us insights in the thematic coherence of fake and LLM-generated news, which is the third research contribution of this work. Beyond its predictive performance, explanations are constructed to help explain the decisions of the proposed method, to bridge the gap between model behavior and user understanding and acceptance. The overall findings highlight the potential for thematic coherence modeling to further advance automated text assessment and detection tools.

Acknowledgements

First of all, I would like to sincerely thank Dr. Davide Dell'Anna for his continuous guidance throughout this thesis project. His great input and critical attitude have helped me tremendously. From answering emails at unorthodox times to ever-late running meetings, the process would not have been the same without his supervision. I would also like to thank Dr. Dong Nguyen for willing to be my second supervisor and her useful feedback during the proposal phase. Next, I would like to thank Tess van Oostrum for her continuous support and the possibility to frequently exchange thoughts on the project. Finally, I want to thank Simon Riezebos and Silvia Fallone, who have helped shape this topic - although it evolved significantly from the initial brainstorms - and provided helpful insights throughout the process.

Contents

List of Figures

List of Tables

1. Introduction

This thesis aims to contribute to coherence modeling by investigating thematic coherence, with a focus on the thematic coherence of fake news and news generated by Large Language Models (LLMs). In section 1.1, we formulate the motivation for this research. Section 1.2 presents the contributions of this work to the existing literature. In section 1.3 we introduce the research questions, and section 1.4 concludes this chapter by providing an outline for this thesis.

1.1 Research motivation

In the early 2000s, the rise of online (social) media platforms profoundly transformed how people communicate, including their means of news consumption. This prompted several concerns, among them that excess variety in opinions would facilitate the formation of "echo chambers" where like-minded citizens would be isolated from contrary perspectives [1]. Nowadays, more people seek out and consume news from online platforms rather than traditional news organizations. The reasons for this shift are inherently tied to the characteristics of such platforms: the high ease of use, the low costs compared to newspapers or television and the high dissemination rate have all contributed to the attraction of millions of users. To illustrate: from 2022 to 2023, the number of EU citizens consuming their news through online media platforms grew by 11 percentage points [2]. While online news consumption offers advantages, it also brings potentially troubling consequences. Owing to the high popularity of online media platforms, large volumes of news are spread each day with high velocity. This forces readers to make quick, face-value judgments on the information they are being presented [3]. As a result, the internet has become an ideal breeding ground for spreading fake news. This includes misleading information, fake reviews, advertisements or forged political statements [4]. All pose a considerable threat to our perception of truth [5].

The period leading up to the US presidential election in 2016 is often regarded in the previous literature as an event that ignited a surge of attention towards the issue of online fake news [6], [7]. During this period, fake news has contributed to increasing political polarization and partisan conflict, influencing voters with misleading or erroneous claims [8]. Throughout this campaign, hundreds of thousands of fake Russian accounts posted anti-Clinton messages such as *'Hillary was a criminal'* or *'Obama had a secret army'*. It was found that 0.1% of Twitter users accounted for over 80% of fake news sources shared, demonstrating the vast power a group of malevolent individuals has in shaping public opinion [9]. The wide reach of fake

news is further emphasized by the fact that during the campaign, the top 20 frequently discussed fake news stories generated more shares (8.7 million) than the most-discussed election stories posted by 19 major news websites (7.4 million) [10]. The recent COVID-19 pandemic additionally illustrated the dangers of fake news, with harmful misinformation like false cures spreading rapidly [11].

The extensive dissemination of fake news can have a far-reaching impact on both individuals and society. As follows from the situations discussed before, fake news can contribute to political polarization or undermine public health efforts, leading to distrust and confusion. Second, it changes how people respond to real news. By regularly being exposed to fake news created to trigger suspicion, their ability to differentiate fake from real news is slowly impeded [12]. On a larger scale, this erosion of trust and the skepticism that arises towards fact-based journalism as a consequence, can harmfully influence people's faith in authorities, experts, government and democracy in general [13].

Given these major implications, it is critical that methods for automated fake news detection are developed. Such methods can then be used to, e.g., enhance content filtering and flagging, support journalists in fact-checking or assist policymakers in designing regulations to combat fake news [12], [14]. However, fake news identification remains a complex issue. A recent study has shown that the power of AI algorithms to identify fake news is significantly lower than its ability to create it [15]. One of the main characteristics of fake news that makes its detection such a challenging task is that its content is designed to closely resemble the truth to deceive its audience. Fake news content varies widely in terms of topics and writing styles and it tends to distort truth by using different linguistic styles. For example, fake news may reference true evidence but place it in an incorrect context to support a non-factual claim [16]. Moreover, fake news tends to have more fragmented stories and far fewer references, further complicating the task for humans to assess the credibility of this content [17]. Another challenge within the domain of fake news detection is the data quality. Fake articles often pertain to newly emerging, time-critical events that have not yet been thoroughly verified due to a lack of supporting evidence or claims [12].

To address these complex challenges, extensive research has been conducted that uses artificial intelligence to detect fake news. This is done along several dimensions, such as knowledge (fact-checking), context (e.g., investigating user profiles) or language styles. Within the latter category, further divisions are made in the existing literature. For example, Sharma et al. [18] showed that on the word-level, fake news articles contain a higher degree of informality (% swear words), subjectivity (% report verbs) and emotion (% emotional words). On a more structural level, it was found that fake news articles thematically deviate between their title/opening sentences and the remainder of the story [6], [19]. Karimi and Tang [20] learned

a hierarchical discourse-level structure and observed that fake news articles have a higher degree of leaf nodes (i.e. isolated sentences), indicating a lower coherence. Similarly, Rubin et al. [21] found an indication that deceptive stories are disjunctive and restate information less frequently, lowering their overall coherence.

Nowadays, with rapid developments related to Large Language Models (LLMs), the task of detecting fake news is further complicated. This is because LLMs significantly enhance the pace with which text is generated. The truthfulness of such texts can vary: either advertently through a malicious actor using the model or inadvertently due to the probabilistic nature of such models which may lead to hallucinations [22]. Given that the popularity of such models surged when OpenAI's ChatGPT was made available to the public [23], it is pivotal to better understand the text generated by such models. Studies have shown that there are properties of LLM-generated text that are distinguishable from human-written text [24], [25]. Therefore, the coherence of LLM-generated text will also be investigated in this research.

While coherence has been analyzed in various domains, its application within the fake news domain remains relatively underexplored. Previous studies within the fake news domain have primarily focused on local coherence, which pertains to sentence-to-sentence relations [20] [26] by tracking e.g. entity progression. However, local approaches may not fully consider the broader context that global coherence provides. Outside the fake news domain, global coherence has been considered [27], but often at the expense of interpretability using highly opaque, deep neural models. This thesis addresses these gaps by developing a robust global coherence model. More specifically, it will focus on the *thematic* coherence, which is an aspect of global coherence that assesses how well a text adheres to its overall theme(s) [28]. It will be tested on a traditional coherence evaluation task and a novel coherence evaluation task specifically designed to assess thematic coherence is proposed. The proposed method will also be applied by analyzing its ability to distinguish real from fake and human-written from LLM-generated news articles. Additionally, explanations are generated that help explain the decisions of the proposed method.

1.2 Contributions to existing work

Overall, this study contributes to the existing literature in three main ways. First, most previous works model local coherence [29]–[31] when studying coherence. A more recent study by Moon et al. [27] highlights the potential of analyzing global coherence patterns. Therefore, a new method to capture global coherence will be developed, specifically focused on *thematic* coherence. Explainability is considered a crucial aspect of this research to bridge the gap between model behavior and user understanding, which is why the method is designed to be

interpretable and also to generate explanations.

Second, to ensure the proposed method captures thematic coherence as intended, it needs to be evaluated at distinguishing coherent from incoherent texts first. Most of the coherence models in the literature are evaluated on the sentence ordering task. Due to its local focus, this task is deemed less appropriate for this study. To address this, a second research contribution is constructing a novel evaluation task for assessing coherence, emphasizing disruptions in thematic coherence over sequential ordering.

Third, we will analyze the usefulness of thematic coherence in determining the veracity - whether an article is real or fake - and the origin - whether an article is written by a human or generated by an LLM. Research on the role of thematic coherence in detecting fake news has been limited and the few existing works have a different focus, e.g. on determining the (in)coherence between the article title and body [19] in real and fake news articles. Similarly, the role of coherence in LLM-generated text has primarily been investigated at the sentence level [32]. The third contribution is therefore an analysis of the differences in thematic coherence between human-written real and fake news and between human-written real news and LLM-generated news.

1.3 Research questions

For this research, we have defined the following research questions.

RQ1 What is a method that can effectively measure thematic coherence in news articles?

Understanding how thematic coherence can be effectively captured is fundamental to this research. Therefore, the proposed method will be outlined in detail. The different modules in the method are designed to be modular and will be discussed individually. Ultimately, the method enhances existing coherence modeling techniques with an interpretable approach to capturing thematic coherence.

RQ2 What is a suitable evaluation task for capturing thematic coherence?

To test whether the proposed method actually captures thematic coherence, a new evaluation task is designed. Most existing research does not take thematic coherence into account and its evaluation tasks are focused on detecting local coherence patterns. To this end, a new evaluation task specifically designed to measure thematic coherence is proposed.

RQ3 How accurately can the proposed coherence method distinguish between human-written real and fake news articles?

To study the third research question, it will be investigated whether there are significant

differences in thematic coherence between human-written real and fake news articles. Additionally, the use of thematic coherence will be tested in correctly classifying news articles as real or fake.

RQ4 How accurately can the proposed coherence method distinguish between human-written and LLM-generated news articles?

Similar to research question three, this research question pertains to the extent to which the proposed method can distinguish between human-written and LLM-generated news articles. With the rapid developments of LLMs nowadays, analyzing and understanding their style is of great relevance.

1.4 Thesis outline

The thesis will be structured as follows. First, a literature review that discusses related works is presented. This literature review aims to sketch the broad landscape while also highlighting the potential significance of the proposed approach and identifying the research gap. In the subsequent chapter, the methodology is described on a conceptual level. After, the new evaluation task is proposed. Following that, the evaluation will be presented: this contains the exact method instantiation and results for each experiment. In the final chapter, the discussion, we will elaborate upon the results, answer the research questions, and consider research limitations and valuable directions for future research.

2. Literature review

Fake news is not a new phenomenon. It has existed for centuries and the spread of invented facts took off at the same time that news began circulating internationally, enabled by the invention of the printing press [33]. Despite its lengthy existence, researchers have not come to an agreed definition of the term 'fake news'. Therefore, it is essential to first discuss and compare some widely adopted definitions of fake news in the existing literature and propose our definition of fake news that will be used throughout this research.

2.1 Fake news definitions

Numerous concepts and terms that refer to fake news can be found in the literature. The majority of terms refer to news articles containing verifiably false information. Aïmeur et al. [34] separate the existing terms into two groups. The first group represents broader, overarching terms such as false information and fake news. The second group represents more specific, elementary terms like misinformation, disinformation and malinformation. While there is consensus on the definitions of the more specific terms, there is still no agreed-upon definition for the terms in the first group. The distinctions can be better understood through Table 2.1, which compares them based on the intent and the authenticity of the news content. In other words, whether the purpose of the article is to mislead or harm and whether its content is verifiably false or not (in which case it is genuine).

Term	**Definition**	**Intent**	**Authenticity**
False information/fake news	Verifiably false information	-	False
Misinformation	False information that is shared without the intention to mislead or to cause harm	Not to mislead	False
Disinformation	False information that is shared to intentionally mislead	To mislead	False
Malinformation	Genuine information that is shared with an intent to cause harm	To cause harm	Genuine

Table 2.1: Comparison between fake news definitions. Table from Aïmeur et al. [34]

As follows from Table 2.1, the general, overarching terms do not have a distinct intent but merely contain verifiably false information. The more specific terms differ with regard to authenticity and intent. However, for the purpose of our research, it is not necessary to make

a hard distinction between the elementary terms. Throughout this research, we adopt a broad definition that encompasses those mentioned before and is also used by Sharma et al. [18]:

Definition 2.1. A news article or message published and propagated through media, carrying false information irrespective of the means and motives behind it.

This definition is opted for because the focus of this research is on understanding the language of fake news, rather than distinguishing between motives like misinformation or disinformation.

2.2 Categories of fake news detection

Having outlined the terminology that will be used in this research, detection methods will now be discussed. In the literature, the detection of fake news is typically divided into two main approaches: context-based detection and content-based detection. First, context-based features and their relevance in detecting fake news will be briefly elaborated upon [12].

2.2.1 Context-based detection

Context-based indicators for fake news can be derived from user-driven social engagements. User-driven refers to the interactions and activities initiated by users on social media platforms. Social engagements reflect the dissemination of news over time, which may provide valuable information to assess the truthfulness of news articles [35]. Some useful examples of such contextual information may include checking if the news and the source that published it are credible, verifying the date and the supporting resources, or whether other news platforms are reporting on the same or similar stories [4]. Generally, a distinction is made between three aspects of context-based detection: users, generated posts and networks.

User-based detection. While many users on social media are legitimate, some accounts may be malicious or not even owned by real humans. Examples of such non-human accounts include social bots that are controlled by an algorithm to automatically produce content and interact with other users [36]. To illustrate the order of magnitude in which such bots are prevalent: a study by Bessi & Ferrara found that around 19 million bot accounts produced one or more tweets supporting either Clinton or Trump in the week preceding the election day in 2016 [36]. Another type of social media user that plays a role in the dissemination of fake news is trolls, real human users who aim to provoke emotional reactions. The same study has shown that 1,000 Russians were paid to spread fake news about Hillary Clinton. Given that such accounts can be influential sources of fake news dissemination, capturing information on the user profile behind a post can assess its credibility. Examples of such information include speaker name or job title, the ratio of followers/followees, profile pictures or political bias [37].

Post-based detection. People express themselves through posts on social media. Post-based analysis focuses on how people's opinions and reactions can be used to infer the veracity of a news article. This can be done in different ways. One option is to analyze individual posts for credibility features to assess the degree of reliability, where a higher credibility implies a higher reliability [38]. Such credibility features include the number of URLs included in the article or the number of re-posts (both positively contribute to the credibility of an article [17]). Alternatively, temporal variations in posting behavior can also be indicative of fake news. Studies have shown that fake news tends to show bursty posting patterns where there is a sudden surge in engagement, followed by a rapid decline. Similarly, irregular posting times at odd hours or in close succession may also be indicative of lower veracity of news articles [39].

Network-based detection. On social media platforms, users often surround themselves with like-minded people and form networks in terms of interests and relations. An example of such a network is a friendship network, highlighting the follower structure of users who place related posts. An extension of this is the diffusion network, which tracks the trajectory of the spread of fake news. Here, nodes are users, and edges represent paths of information diffusion [40]. Once properly built, network metrics such as diffusion path length or speed of propagation can be used to enhance insights into the spreading patterns of fake news.

Even though these context-based approaches can enhance the understanding of the propagation of fake news, which helps in detecting it, there are some drawbacks. When it comes to user-based fake news identification, it is considerably platform-dependent. Different platforms will have varying demographics and behaviors, making it hard to implement a one-size-fits-all approach [41]. Also, malicious actors can more easily manipulate the creation and behavioral patterns of user profiles which can be automated to mimic legitimate user behavior [42]. Even though content can be manipulated too, ensuring a text stays coherent, contextually appropriate and factually plausible is more difficult than managing fake user profiles [43]. Post-based analysis relies heavily on user engagement data like shares and comments. If an article fails to provoke strong reactions, it may not provide enough data for useful post-based analysis. Moreover, in the presence of sufficient reactions, they tend to be rather noisy, which can complicate analysis [44]. Network-based approaches also require monitoring the propagation of (fake) news, which can delay the detection process. This delay can be critical, allowing information to be widespread and causing harm before it has been identified [45]. Overall, the three approaches are highly tailored to social media platforms and are often used as auxiliary data instead of on their own [35]. A more direct analysis of the primary source of information - that is, the text of an article - provides a platform-independent approach. Therefore, content-based detection will be expanded upon in the next section.

2.2.2 Content-based detection

The second strategy that is used in the identification of fake news is content-based detection. Content-based detection investigates the text, linguistic patterns and writing styles for both real and fake news, to then capture the most discriminative features for fake news detection. We categorize content-based analysis into two main pillars: knowledge-based and language-based.

Knowledge-based detection. The first category that we will discuss is knowledge-based analysis, which refers to directly checking the truthfulness of major claims in news articles. This is done by websites such as Politifact.com, which assign a binary or ordinal truth value to (political) statements. For example, Biden stated in an interview on May 15, 2024 that "Violent crime is near a record 50-year low" [46]. This is given the label 'true' based on crime rates provided by the FBI: only 2014 and 2019 have had lower crime rates in the last 50 years. However, in a speech a day earlier, Biden said "Inflation was 9% when I came to office", which is labeled as 'false': when he was inaugurated, year-over-year inflation was about 1.4% [46]. Fact-checking is crucial for maintaining the integrity of information in public discourse. Therefore, automated fact-checking has received significant attention over recent years. Automated fact-checking mechanisms often follow a similar pipeline: detecting the claim from a spoken or written statement, retrieving evidence that supports or refutes this claim, basing a verdict prediction on the (dis)similarity between the claim and the evidence and ideally also providing a justification for the prediction [47] [48]. A main concern in such knowledge-based techniques however, is obtaining a high-quality database from which to retrieve evidence, as news tends to evolve over time and be diverse in terms of topics, purposes and styles. Zeng et al. [49] identify some other pressing challenges in automated fact-checking. First, it is complex to determine the conceptual definition of a claim, as it depends on the interpretation of 'check-worthiness'. Second, annotation issues are likely as fact-checking websites only list claims, rather than non-claims, which means that one needs to develop models that only leverage instances of the positive class. Third, datasets are often highly imbalanced as not check-worthy claims far outnumber check-worthy claims, which may result in overfitting. Finally, it requires dependence on (multiple) external databases that would need to be updated regularly to stay up-to-date.

Language-based detection. Language-based analysis encompasses the study of language at the lexical, syntactic or semantic level. These levels are classical areas of studies within natural language processing (NLP) [50]. Analyzing them can be particularly useful for tasks such as identifying the writing style of an author, which is important in uncovering potentially deceptive information: malicious online accounts tend to express deceptive information by intentionally obfuscating their writing style or attempting to imitate other users [51]. At each

level, we can identify features that may be indicative of fake news. Some features will briefly be highlighted next, including their role in detecting fake news.
The first linguistic features are lexical features. This refers to the analysis of lexicons at the character- or word level. The main task at the lexicon level is assessing the frequency statistics of lexicons, which can basically be conducted using a bag-of-words (BoW) model [10]. A bag-of-words model essentially transforms text into absolute frequency counts [52], which can then be used for analyzing frequencies of words indicative of deception. Example features of a news article include the characters per word, the number of total words, or the frequency of difficult, emotionally charged or unique words. To illustrate: fake news text has been found to e.g. have higher diversity (% of unique words) or be more emotional (% of emotional words).
Second, text can be analysed using syntactic features. These refer to the rules and principles that govern the structure of sentences [53]. Examples include sentence-level features such as the frequency of certain types of words or phrases, which can be captured using e.g. a part-of-speech (POS) tagger. Research suggests that fake news e.g. contains fewer proper and common nouns but a higher usage of personal pronouns [54].
Thirdly, text can be analyzed at the semantic level. Studying semantics involves understanding the meaning of text while taking context and logical structuring of sentences into account [55]. This can e.g. be done by calculating the semantic similarity between adjacent sentences to track the progression of ideas: real news texts tend to have a smoother, more coherent progression of ideas [56].
This language-based detection is the primary focus of this thesis and section 2.3 will provide a more in-depth analysis of language-based detection methods.

2.2.3 Hybrid approaches

The majority of researchers focus on either content-based or context-based methods for fake news detection due to challenges in combining these approaches, such as feature correlations and semantic conflicts [57]. However, more recent efforts intend to use a mixture of both news content-based and social context-based approaches. An example includes combining information from the publisher with semantic and emotional information in the news content [58]. Even though hybrid approaches provide a promising research avenue, this research will focus on content-based and, more specifically, language-based detection. This approach can potentially be embedded in hybrid approaches in future work.

2.3 Language-based detection methods

In this section, an overview of language-based detection methods is provided. The underlying basis of language-based detection is that textual content in fake news differs from that in

true news in some linguistically quantifiable way [12][10], which was touched upon in section 2.2.2. In order to determine veracity, language cues indicative of fake news can be identified using more traditional hand-engineered word-based methods, structure-based methods or more complex deep learning methods [18]. This section will further elaborate upon language-based methods to detect fake news.

2.3.1 Word-based detection methods

Given the high number of features that can be used in analyzing text, not all word-based features that may be indicative of fake news will be discussed here. Instead, these features can be grouped into psycho-linguistic categories. In this research, we draw on the categorization of Zhou & Zafarani [10] and discuss some corresponding computational features. In doing so, we focus primarily on the features that have been shown to reveal patterns of fake news that are distinguishable from true news [59] [44].

Complexity. Zhou & Zafarani define several features within the complexity category. Some of them include the number of words per sentence, the number of technical words and the number of quotes and punctuations. Another way in which word complexity can be captured is by using grade-level readability indexes, like the Gunning-Fog or the Flesh-Kincaid index. Using these features, Horne and Adali [6] have shown that fake news articles use shorter words, fewer punctuation and quotes, and require a lower educational level to read (as measured by the readability indices). Moreover, fake news tends to use fewer technical and analytical words [6].

Subjectivity. The second category is the degree of subjectivity in news articles, measured by the amount of subjective verbs (e.g. 'feel', 'believe'), the amount of biased lexicons (e.g. 'attack') and the number of report verbs (e.g. 'announce'). Zhou et al. [10] found that fake news articles contain a higher number of report verbs and subjective verbs.

Sentiment. Sentiment analysis has received considerable attention in the domain of fake news detection. This can be captured through e.g. the number of positive/negative words or the number of anxiety/anger/sadness words. Extreme sentiment expressed in a news headline tends to indicate a higher degree of sensationalism, which is more prevalent in fake news articles [59]. Kapusta et al. found that fake news articles have a more negative sentiment and a higher degree of emotional words. Such language in articles is more likely to capture a reader's attention and evoke stronger emotional reactions, making its content more memorable and likely to be shared [60], which is why fake news creators may willingly opt for such a style.

Diversity. The extent to which the content of a news article is diverse has also been proven to be an indicator of its veracity. This can be quantified through measuring e.g. redundancy

(the number of unique function words) or the type-to-token ratio (TTR), which is the number of unique words divided by the total number of words in the document. Fake news articles have been found to have a higher degree of redundancy, indicating they are filled with less substantial information [6].

Informality. The final category discussed here is the degree of informality, as measured by e.g. the number of typos or swear words. It has been observed that fake articles contain a higher percentage of both typos and swear words [59].

These word-based methods have proven to be useful in revealing patterns present in fake news articles. However, they have notable limitations. These methods typically focus on word frequency and treat individual words equally without taking the syntactic or semantic roles into account. As a result, they tend to overlook the deeper contextual relationships between words and phrases that contribute to the overall meaning of an article. Besides, these hand-crafted features are often tailored to specific domains or datasets, limiting their generalizability to other settings. Moreover, with generative AI becoming increasingly more competent, differences between real and fake news are likely to become less pronounced [61]. Therefore, more structure-based techniques are required that can look beyond the surface level. These will be discussed in the next section.

2.3.2 Structure-based detection methods

To better capture how information in a text is organized and presented, it is relevant to explore structure-based methods. These approaches highlight the syntactical and grammatical features of text, which can be used to distinguish fake from real news. Two widely used methods in NLP will briefly be discussed.

Part-of-Speech tags. Part-of-speech (POS) tags are obtained by tagging each word in a sentence according to its syntactic function (nouns, pronouns). The frequency of POS tags has previously been found to be linked to the genre of the text being considered [18]. Ott et al. [62] tested whether this variation also persisted with respect to text veracity. Their analysis showed that deceptive articles contain more verbs, pronouns and pre-determiners, whereas truthful articles contain more nouns, adjectives and coordinating conjunctions [62].

Probabilistic Context Free Grammar. Later works have also investigated deeper syntactic features derived from probabilistic context-free grammars (PCFG) trees. A Context-Free Grammar (CFG) tree represents a sentence's grammatical structure, with words as terminal nodes and syntactic elements like noun phrases as intermediate nodes. A probabilistic CFG (PCFG) disambiguates multiple possible structures by assigning probabilities to trees based on the likelihood of production rules, which are statistically derived from a corpus [18]. Feng et al.

[16] investigated the use of PCFGs to encode deeper syntactic features for deception detection. They trained an SVM classifier and found that PCFG features are more successful than POS tags in classifying fake texts. In analyzing the most discriminative constituents, it was observed that constituents like indirect inquiry, verb phrases and conjunction clauses were more prevalent in fake texts compared to true ones [16].

POS tagging and PCFGs offer a more refined analysis of textual structure by examining syntactic patterns. However, these approaches are limited in their ability to capture context-sensitive information across sentences [18]. To capture the flow of an entire document, we must look beyond these localized patterns to consider how well a text adheres to the central theme. This is where coherence-based methods are particularly useful, as they enable an evaluation of both the local and the global structure of a text. This is important for the classification of longer (fake) news articles, where the methods above have so far shown limited effectiveness [18].

2.3.3 Coherence-based detection

Coherence is considered an important quality of effective writing that enhances readability and allows a message to be conveyed in a meaningful way [63]. In the Cambridge Dictionary, coherence is defined as follows: "The situation when the parts of something fit together in a natural or reasonable way" [64]. Evidently, this is a very broad definition of coherence that still leaves ample room for interpretation. A linguistics paper by Lee [65] adopts a more narrow definition, describing coherence as the relationships that link the ideas in a text to create meaning for the readers. However, they also admit that this is still a fuzzy concept. To come to a more workable definition, let us examine how it is defined in the context of fake news detection. Within this domain, the term coherence is often used to denote the overall consistency of a document in adhering to its core focus theme(s) or topic(s). Similarly, it can be considered as opposite to the notion of dispersion or scatter: a document that frequently switches between themes lacks coherence [28]. A more precise definition is given by Bruce [66]: Textual coherence refers to the quality that makes a text logically organized and semantically consistent, ensuring that the ideas and sentences are well-connected and flow smoothly. It involves the logical arrangement of words, phrases, and sentences to create a clear and understandable narrative.

Coherence has, albeit not extensively, also been explored in the context of fake news detection [19] [28]. Previous work suggests that coherence may be a valid indicator of the veracity of a story and thus helpful for detecting fake news. Rubin et al. [21] unveiled this by applying Rhetorical Structure Theory (RST) [67] to study the discourse of (deceptive) stories posted online. They found that a critical distinguishing factor within deceptive stories is that they are

disjunctive. Also, they found that truthful stories provide more evidence and restate information more frequently, positively contributing to their coherence.

Madani et al. [26] aim to detect fake news using a variety of features. They identify different types of linguistic features, as discussed in section 2.3.1: surface features (number of words/sentences, number of adjectives/adverbs) or text polarity features (scores given based on the presence of negatively or positively charged words). They add to this two features: coherence and cohesion, with the latter further divided into grammatical and lexical cohesion. The grammatical cohesion is measured by calculating the number of reference pronouns and conjunctions. The lexical cohesion is measured by converting each sentence into a numeric vector and then calculating the cosine similarity between adjacent sentences. Coherence, on the other hand, is calculated by measuring the semantic similarity between the sentences at the beginning and the end of each news sample (it is argued that the first and the last sentence form the topic sentence and conclusion, respectively, and that the semantic relationship between the remaining sentences is captured in cohesion). The study found that adding the two features improved the performance of their fake news detection model.

Anspach [68] found that an increasing amount of readers only skim through an article. Either because they overestimate their political knowledge or because they are only in search of some hasty form of emotional affirmation, which presents malicious actors the opportunity to deftly intersperse news with falsity. Building on this, researchers have investigated the coherence between the beginning and the end of an article. Some have focused on spotting incongruences between the headline and the body text [69], [70] and found that misleading headlines often present overrated or false information not supported by the body text. Dogo et al. [19] extended their analysis beyond only headlines, investigating whether fake news articles thematically deviate between their opening sentences and the remainder of the story. They do so by calculating the Chebyshev and Euclidean distances to measure topical divergence, whici is defined as the difference between topic probability distributions for parts x (the first one to five sentences) and y (the remainder). They extract these topic probability distributions using Latent Dirichlet Allocation (LDA) as topic modeling method. This paper is among the most comparable to our study, given the methodological similarities with the method proposed in this thesis - particularly in assessing coherence through topic modeling. Therefore, topic modeling will be discussed separately in section 2.3.4, and how the method proposed in this thesis diverges from and improves upon the method proposed by Dogo et al. will be further discussed in section 2.6. Dogo et al. measure topical divergence between the opening sentences x and the remainder y as:

- Chebyshev (D_{Ch}): $D_{Ch}\left(p_{xi}, p_{yj}\right) = \max_i \left|p_{xi} - p_{yj}\right|$

- Euclidean (D_E): $D_E\left(p_{xi}, p_{yj}\right) = \left\|p_{xi} - p_{yj}\right\| = \sqrt{\sum_{i=1}^{m}\left(p_{xi} - p_{yj}\right)^2}$

Here, for $i = (1, \ldots, m)$ topics, $p_x = (p_{x1}, \ldots, p_{xm})$ and $p_y = (p_{y1}, \ldots, p_{ym})$ are two vectors of topic distributions, which denote the prevalence of a topic i in the opening text x and remainder y of an article, respectively. Intuitively, the Chebyshev distance is the greatest difference found between any two topics in x and y, whereas the Euclidean distance measures how "far" the two topic distributions are from one another. When comparing the first five sentences to the remainder of the article, they found the greatest disparity between real and fake articles. Under this condition, the Chebyshev and Euclidian distances were consistently lower for real than for fake articles. This shows that fake news articles show a greater and significant thematic deviation between their opening sentences and the remainder of the article.

Coherence has alternatively been investigated in the literature by analyzing the discourse composition of real versus fake news articles. Karimi & Tang [20] incorporate the hierarchical discourse-level structure of real and fake news articles. The hierarchical-discourse level structure for fake news detection (HDSF) framework they propose learns and constructs this structure by using a Bi-Directional Long-Short Term Memory (BLSTM) network to all sentences (which act as the discourse units) of a document to obtain a sequence of sentential representations $f_1, f_2, \ldots, f_k$ (see also Figure 2.1). They proceed to construct and optimize an inter-sentential attention matrix $\mathbf{A} \in \mathbb{R}^{k \times k}$ containing the parent-child probabilities between the sentences to identify the dependency between two sentences. Following this, they calculate the probabilities for all sentences of that sentence being the root. Using $\mathbf{A}$ and the array of root probabilities, the discourse dependency tree can be constructed: this is done by applying a greedy algorithm starting at the sentence with the highest probability of being the root. From there, at each iteration, the maximum entry in the submatrix of A is found, which is formed by the rows of the current nodes and the columns of the remaining nodes.

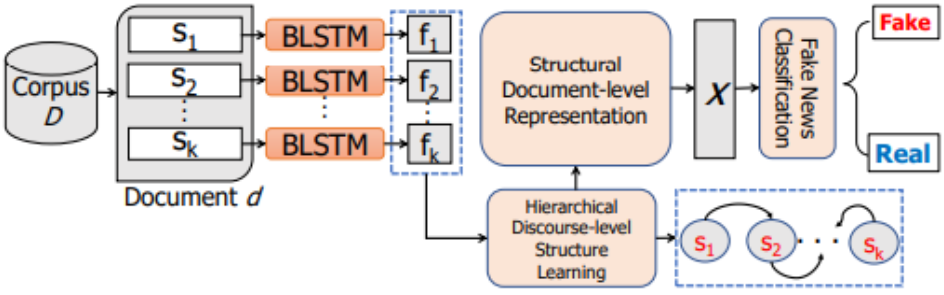


Figure 2.1: The HDSF framework (image by Karimi & Tang [20])
[20]

Karimi & Tang [20] identify a couple of insightful structure-related properties of fake news that explain the learned structures. Their purpose of this is twofold: first, they intend to high-

light how fake and real news articles are different, and second, they aim to leverage these properties to gain insight into the coherence differences in both types of articles. They define the following three features:

1. Number of leaf nodes: $P_l = \frac{l}{\log(k)}$, with l the number of leaf nodes and k the total number of sentences in a document;
2. The preorder difference: $P_t = \frac{\sum_{j=1}^{k} \left| s_j^{\text{position}} - j \right|}{\log(k)}$, with s_j^{position} denoting the position in the preorder traversal of the tree of a document;
3. The parent-child distance: $P_c = \frac{\sum_{\forall c,p \in \mathcal{T}} \left| c_{\text{position}} - p_{\text{position}} \right|}{\log(k)}$, where p and c denote parent and child respectively.

The rationale behind the three properties is as follows. First, a higher degree of leaf nodes (i.e. isolated sentences) means fewer sentences are linked and implies a lower coherence. Second, sentence order is closely linked to the coherence of a document where a displaced order (high pre-order difference) reduces coherence [71]. Third, one would expect the child node to be close to the parent node in its original sequential order, such that a high distance indicates a weaker coherence. They find that in all three properties, the real documents show a significantly lower value than the fake documents, demonstrating that real news documents exhibit a higher degree of coherence.

Singh et al. [28] quantify lexical coherence in a different manner, namely by building upon advancements in three different directions within NLP. Before discussing their computational approach, we briefly outline each of the NLP building blocks that form the basis of the coherence assessments:

- Text embeddings. They refer to techniques that convert text data into numerical vectors that are processable by machine learning models. Such techniques, like GloVe [72] and word2vec [73], map each word in the document corpus to a vector of a pre-specified dimensionality by considering the words' lexical proximity within the documents where they occur.
- Explicit semantic analysis (ESA). In ESA, structured knowledge bases like Wikipedia are used to derive meaningful representations of text. ESA represents text as vectors in a high-dimensional space, where each dimension corresponds to a Wikipedia concept [74]. It is often used for applications related to computing semantic relatedness, Singh et al. [28] use it to estimate document-level coherence.
- Entity linkings. Entity linking methodologies [75] aim to directly link fragments of text to specific entities in knowledge bases like Wikipedia. In short, a text document is converted to a set of Wikipedia entities referenced within it to provide a semantic grounding

for measuring lexical coherence.

Having defined this, Singh et al. [28] modeled coherence as the mean of pairwise similarities between elements of the article. Elements are either sentences (for text embeddings and ESA) or entities (for entity linking). The sentence-level coherence is calculated as:

- Sentence-level coherence (C_{sent}): $C_{\text{sent}}(D) = \text{mean}\left\{\text{sim}(\text{rep}(s_i), \text{rep}(s_j)) \mid s_i, s_j \in D, i \neq j\right\}$

Whereas the entity-level coherence is calculated as:

- Entity-level coherence (C_{ent}): $C_{\text{ent}}(D) = \text{mean}\left\{\text{sim}(e_i, e_j) \mid e_i, e_j \in D, i \neq j\right\}$

Here, D is the document being analyzed, and $\text{sim}(\text{rep}(s_i), \text{rep}(s_j))$ is the function measuring the similarity between the representations of sentences s_i and s_j (or entities e_i and e_j). Using these concepts, they quantify the document coherence using either sentence-level or entity-level modeling. For the sentence-level modeling, coherence is quantified by averaging the word and ESA vectors within a sentence (for the embeddings and ESA approach respectively). For the entity-level, Wikipedia2vec vectors are used to represent entities in the text and its average is calculated to assess coherence.

They found that in each of the three coherence assessments - across two datasets covering different domains - the fake news articles were found to be less coherent, with the differences being statistically significant in 5 out of 6 combinations. They also observed that using the embedding mechanism is the most suited to discern differences between real and fake articles.

2.3.4 Topic Modeling

As outlined, the topic modeling approach taken by Dogo et al. [19] provided inspiration for the way thematic coherence is modeled in this research. Therefore, it is crucial to discuss how topic modeling works and how it can be applied in the coherence modeling domain.

Topic modeling is a statistical type of modeling used to identify topics within a (collection of) document(s). Different approaches for extracting topics from text are available in the literature [19] [76]. Latent Dirichlet Allocation (LDA) is a more traditional method, which is a generative probabilistic model that aids the discovery of latent themes or topics in a corpus [77]. Latent in this context refers to the underlying structures in the data. LDA however relies on co-occurrences of words across many documents to find patterns and form topics. When applying LDA to a single article (as is the case in this research), the model has too limited information to learn a robust topic distribution, which results in repetitive and uninformative topic distributions as it needs a large corpus to extract topics from. More recently, transformer-based models like BERT have proven to be a very powerful solution for this issue [78]. Taking a deep-learning-based approach over a word frequency approach and focusing on contextual

embeddings allows these models to derive more meaningful topics for shorter texts. A topic modeling technique that leverages transformers is BERTopic [79]. A brief overview of how BERTopic extracts topics from a dataset is outlined below.

BERTopic can be viewed as a sequence of 6 steps to come to its topic representations. Given its central role in extracting topics from the analyzed articles in this research, a concise explanation of the steps on a conceptual level is provided below. The exact configuration used in this research is further expanded upon in 3.1.2, as BERTopic is very modular and allows different options for each step (illustrated in Figure 2.2).

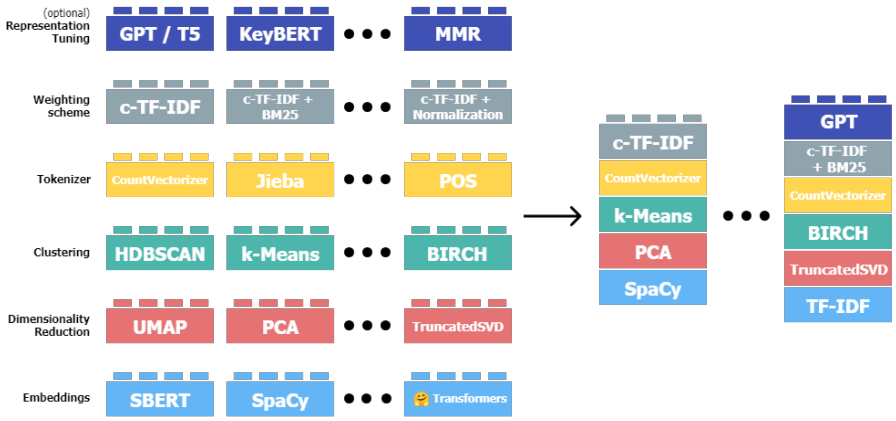


Figure 2.2: BERTopic algorithm.

1. **Embeddings.** Models analyzing natural language require numerical representations of documents in order to process them. This is typically achieved through embeddings. Embeddings are vector representations that provide numerical mappings of text. BERTopic allows for multiple methods to create embeddings, such as ones optimized for multiple languages or for semantic similarity.

2. **Dimensionality reduction.** Cluster models typically have issues with handling high dimensional data due to the curse of dimensionality, where higher dimensions quickly increase computational complexity and sparsity. Therefore, the dimensionality of the input embeddings needs to be reduced to a workable dimensional space.

3. **Clustering.** After having obtained the reduced embeddings, clusters can be formed based on similar embeddings. Groups of similar embeddings will be the basis of extracting the topics.

4. **Vectorizers.** In topic modeling, creating meaningful topic representations is crucial for interpreting and understanding the discovered patterns. Vectorizers play a key role in this process by transforming textual data into numerical formats, enabling the identification of key terms or n-grams that best represent each topic. These representations are flexible and can be adapted based on the use case.

5. **Topic representation.** To get a good representation of the topics from the bag-of-words matrix, BERTopic uses TF-IDF. TF-IDF (Term Frequency-Inverse Document Frequency) is a statistical measure used to evaluate the importance of a term within a document relative to a corpus, where term frequency (TF) measures how often a word appears in a document, and inverse document frequency (IDF) reduces the weight of terms that are common across the corpus [80]. In BERTopic, TF-IDF is adjusted to work on a cluster level: c-TF-IDF. For a term x within class c, the weight $W_{x,c}$ is calculated as:

$$W_{x,c} = \|\text{tf}_{x,c}\| \times \log\left(1 + \frac{A}{f_x}\right)$$

where $\text{tf}_{x,c}$ is the frequency of word x in class c, f_x is the frequency of word x across all classes, and A is the average number of words per class. So, after converting each cluster to a single document, the frequency of word x in class c is extracted. This results in the cluster-based tf representation that reflects the importance of a word in the cluster. Then, the idf representation is calculated to downweight words that frequently appear across many clusters to highlight words that are more specific to individual clusters.

6. **(Optional) Fine-tune topics.** After having generated the c-TF-IDF representations, BERTopic has obtained a set of words that describe a collection of documents. c-TF-IDF is a method that can quickly generate accurate topic representations. However, with the fast developments in the NLP world, additional fine-tuning of the topic representations can be opted for and can be done using techniques like e.g. GPT, T5 or KeyBERT.

2.4 General coherence modeling

Coherence-based detection techniques, including those leveraging topic modeling, provide valuable insights. However, to more comprehensively understand the literature on coherence, we must also explore the broader context of coherence modeling. Coherence has for a long time been a topic of interest within the broader domain of text analysis. In 2004, McNamara et al. [81] were one of the first to come up with a highly comprehensive framework to systematically use coherence features for automated text analysis. They developed Coh-Metrix, a computational tool that analyzes texts using over 200 measures of coherence, readability and language. It covers a broad spectrum of hand-crafted features, ranging from lexical diversity to syntactic complexity, referential cohesion and readability scores. Even though Coh-Metrix provided a great baseline for coherence measurements, advancements in the field of NLP have paved the way for more complex coherence analysis techniques.

Having primarily focused on coherence applied to fake news detection, it is also important to consider a couple of other advancements in coherence modeling outside the fake news domain. This will help shape our understanding of the different dimensions of coherence and how they have been represented computationally. Many coherence models are namely not applied to fake news detection but to benchmark coherence evaluation tasks like sentence ordering or readability assessments [29] [82]. In sentence ordering, a text is compared with random permutations of its sentences [29], where a coherence model should rank the original text higher than its permutations in terms of coherence. Readability assessments evaluate the ease with which a document can be read and is measured by combining traditional readability indices with statistical language models and syntactic analysis [83]. Many approaches to local coherence modeling rely on entity relations between sentences. The entity grid [29] and entity graph [30] are well-studied frameworks for representing entities in a text. They use grids and graphs respectively to capture entity relations across sentences. An entity grid is a 2 dimensional array with the rows representing sentences and the columns representing discourse entities. This way, syntactic roles (e.g. subject, object) can be tracked over sentences. Local coherence is then captured by means of entity transitions. To make this representation workable for machine learning algorithms, Barzilay and Lapata [29] compute the transition probabilities and generate feature vectors representing the sentences. They achieve an accuracy of ~85% and ~84% on sentence ordering and readability tasks respectively. They however identify some disadvantages, such as data sparsity and computational complexity.

In order to overcome these issues, the entity graph was proposed by Guinaudeau and Strube [30]. A graph can span an entire text (instead of being restricted to transitions between adjacent sentences) without computational complexity or data sparsity problems. They represent text as a bipartite graph with one set of nodes corresponding to sentences and another corresponding to entities mentioned in those sentences. Edges between nodes are created if a sentence contains a particular entity (possibly also including weights assigned based on the grammatical role of the entity in the sentence). The local coherence of a text is then calculated by projecting this bipartite graph into a one-mode graph where nodes represent sentences connected by shared entities. Then, the average outdegree is used as centrality measure to assess how well-connected the sentences are in terms of shared discourse entities. This results in accuracy values of 0.889% and 0.766% for sentence ordering and readability assessments tasks respectively [30].

Building on this, other methods have been proposed to enrich these coherence representations. Recent research has highlighted the effectiveness of using convolutional neural networks (CNNs) for extracting features from entity grids/graphs to encode coherence [84]. Mesgar et al. [31] have used this to revisit graph-based coherence assessments by introducing a neu-

ral graph-based local coherence model. They represent text via a graph, given its ability to capture long-distance relations. Such graphs (see also Figure 2.3) contain two types of edges: edges capturing entity-based relations between sentences and edges capturing the linear order of sentences. They encode these graphs in a different manner, namely via a Relational Graph Convolutional Network (RGCN) [85]. RGCNs encode graph nodes into vectors using the graph's connectivity structure and other information features, with a self-attention layer applied to these vectors to assess each sentence's contribution to overall coherence. This is finally summarized into a coherence score. See Figure 2.3 for an overview of their model.

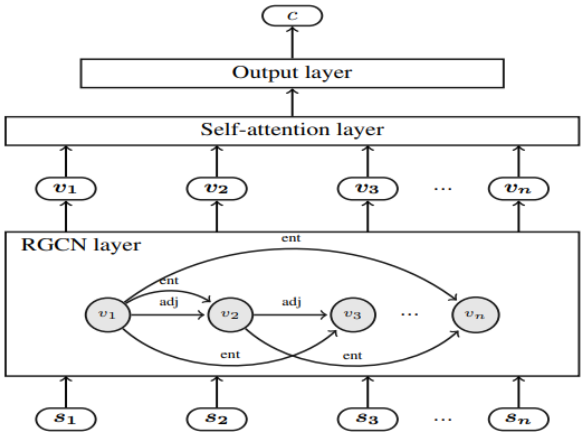


Figure 2.3: Overview of the coherence model (image by Mesgar et al. [31])

Their results show that the neural graph-based local coherence model outperforms neural grid-based local models [84] [86] by about 3.1% for sentence ordering and 1.2% for summary coherence rating. Moreover, their model performs on par with a more recent [27] while using 50% fewer parameters.

Finally, a unified neural coherence model was proposed by Moon et al. [27]. Although they move away from the graph-based approach, it is relevant to discuss as their coherence model incorporates sentence grammar, inter-sentence coherence relations and global coherence patterns into a unified neural framework. It usefully highlights the difference between local and global coherence. Local coherence operates at the sentence level, ensuring smooth (entity) transitions and logical connections between sentences. Global coherence operates at the document level, looking at e.g. consistency of topic progression. Whereas many graph-based approaches adopt a local approach [30] [31], their approach uses a unified model that captures coherence along three dimensions. First, they model the syntactic structure of a sentence. The sentence structure is modeled using a bi-directional LSTM that encodes each word into a contextual representation, capturing both forward and backward dependencies in the sentence. The model is then trained using a language model loss to ensure the representation reflects both meaning and grammatical structure. Second, the relations between sentences are

modeled using the representations from the bi-LSTM for two consecutive sentences, to which a learnable tensor is applied. Finally, Moon et al. [27] also model global coherence patterns, which is where it diverges from the majority of previous research into (local) coherence modeling. They model global coherence by passing all generated sentence representations through convolutional layers with residual connections, followed by an average pooling layer. By taking all sentences as input, patterns of entity distribution and topic progression are considered throughout the entire document. The document-level features are then integrated with the local coherence features to come to a comprehensive coherence score [27].

2.5 Detecting LLM-generated text

Developments related to Large Language Models (LLMs) capable of producing high-quality texts give a new dimension to all tasks involving text analysis nowadays. For news creation, journalists may be inclined to use it as an assisting tool in writing articles. However, the high accessibility of such models may also provide malicious actors the opportunity to quickly create fake articles that may appear real at first glance [61]. Even though this is only one example, it already gives insights as to why it is crucial to gain a better understanding of the characteristics of LLM-generated text. In this section, the scope is limited to the use of LLMs for news generation. The use of LLMs affects news generation both advertently and inadvertently. It can advertently be used by malicious actors to quickly create articles with misleading or fabricated content that mimics legitimate reporting styles. However, LLMs can also have an impact on the truthfulness of news due to their propensity to generate content that is inaccurate or lacks factual precision [42]. This can be due to imperfect training data but is also inherent to the probabilistic nature of LLMs, which can lead to hallucination, where the model generates inaccuracies with no malevolent intent [87]. Therefore, regardless of intent, it is important to address the impact of LLMs on the news domain. This leads to the question to what extent LLM-generated text can be recognized. One study found that the initial accuracy of human labelers on detecting human vs. GPT-3 output text was similar to chance [88]. Even training the labelers on the task at hand only improved accuracy to 55%. Other studies confirm this near-chance performance of humans on detecting LLM-generated text [89]. This illustrates the importance of improving automated detection methods.

LLM-generated text detection is however a challenging task. A recent study has even stated that as LLMs become better at mimicking the distribution in human text, reliable detection will become increasingly more difficult and inevitably impossible at some point [90]. On the contrary, a wide range of other studies assume that human-written text has properties that are distinguishable from LLM-generated text. To limit the scope of this section of the literature review, we will logically only focus on coherence-related properties.

Fröhling et al. trained detectors with one of the most complete sets of features for LLM-generated text detection [22]. Their features can be categorized into four kinds of errors LLMs typically make: lack of syntactic diversity, lack of purpose, repetitiveness of words and lack of coherence. Related to the latter characteristic, they found that language model generations are often surprisingly fluent at first read but lack coherent thought and logic on closer inspection [22]. Closely related and also highly relevant for this research is the topic drift, where LLMs struggle to focus on a single topic and instead cover multiple, unrelated topics in a single text [24]. Badaskar et al. use the topic redundancy, measured by the information loss between a text and its truncated form, as a measure of coherence and hypothesize that human-written text is more redundant as it coherently treats a single or a few topics [24]. Another study found that LLM-generated text is coherent at the sentence level, but perform worse on paragraph-level coherence scoring. This is due to LLM-generated text showing lower relevance, which is defined as the degree to which sentences are relevant to the underlying discourse topic. They attribute this to the token-level training objective of LLMs, which optimizes the prediction of the next word but is not explicitly designed to ensure consistency across longer contexts [25]. This further illustrates that investigating the global coherence of LLM-generated text is a promising research direction.

2.6 Research gap

This literature review glanced over several important topics surrounding coherence modeling in the fake news domain, in the LLM domain, and in general text analysis. Evidently some levels and combinations of analyses are still missing in the existing literature. Table 2.2 provides an overview of the most relevant discussed papers that helps to unveil overlooked elements. The papers are analyzed using three criteria: first, whether they focus on local, sentence-to-sentence coherence or whether they focus on global, document-level coherence. Second, thematic coherence is added in addition to global coherence, as some papers may model (global) discourse flow without explicitly modeling thematic coherence. The third and final criterion is the application of the coherence model.

Taking all of the literature discussed and the overview of Table 2.2 into account helps to identify the research gap. The first research opening, relating to the first research contribution, is the creation of a new, interpretable method to model thematic coherence. Thematic coherence has rarely explicitly been modeled. Dogo et al. address this to some extent, however they define it as the thematic agreement between the first sentences and the remainder of a text [19]. This thesis improves upon this research by providing a more holistic approach (considering the progression of themes over multiple text segments), introducing an evaluation task to more thoroughly test the model, using more flexible divergence metrics (rather

Reference	Local/Global	Thematic coherence	Application
Barzilay & Lapata [29]	Local		General
Strube & Guinaudeau [30]	Local		General
Mesgar et al. [31]	Local		General
Moon et al. [27]	Both		General
Karimi & Tang [20]	Local		Fake news
Singh et al. [28]	Local		Fake news
Dogo et al. [19]	Global	✓	Fake news
Madani et al. [26]	Local		Fake news
Badaskar et al. [24]	Global		LLM-text
Frohling et al. [22]	Both		LLM-text

Table 2.2: Overview of existing coherence modeling methods.

than a static distance), and using a state-of-the-art topic model. The second research contribution is a new evaluation task. The majority of the discussed papers do not focus on thematic coherence [29], [30], [82]. Therefore we introduce a new task specifically designed to assess thematic coherence, thereby extending the pool of existing coherence evaluation tasks. The final research opening is related to the application of the proposed method and pertains to the third research contribution. None of the studies encountered use a (thematic) coherence model to gain insights into the language of both fake and LLM-generated news articles.

3. Proposed Thematic Coherence Computation Method

This section presents our end-to-end method for assigning a thematic coherence score to natural language texts. Proposing such a method is the first research contribution, as stated in section 1.2. Figure 3.1 graphically displays the procedure, referred to as the thematic coherence computation procedure in the remaining sections. It is important to outline the definition for thematic coherence used in this research again, in line with the definition used by [28]:

Definition 3.1 Thematic coherence is the ability of a text to stay focused on its main theme(s).

The following sections discuss the functionality of the different steps on a conceptual level. It must be noted that this pipeline is intended to be modular, allowing the exploration of alternative methods at each step. The procedure contains two phases: phase one is the topic extraction phase, and phase two is the thematic coherence computation phase. Note that the phases are two distinct processes: the extracted topics can be stored and reused later to analyze an individual article. In Figure 3.1, each rectangle depicts a module that contains functionality that could be altered in future implementations. For example, the way topics are assigned, depicted by the 'topic assignment' module in Figure 3.1. Each rounded rectangle represents the input/output of the modules to which they are connected. For example, the 'compute divergence' module has the topic probability distributions for the full article and each chunk as input and outputs divergence scores between the distributions of the full article and each chunk. The arrows connect the modules and their corresponding inputs/outputs. The bottom half of the pipeline includes two distinct paths represented by dashed and solid lines. For elements connected by two arrows, dashed arrows pertain to processes involving chunks, while solid lines refer to the full article. Note that a single arrow signifies that the input/output combines the chunks and the full article, such as for the output of the 'compute divergence' module. The modules will be discussed individually for both phases, preceded by the data and data pre-processing requirements.

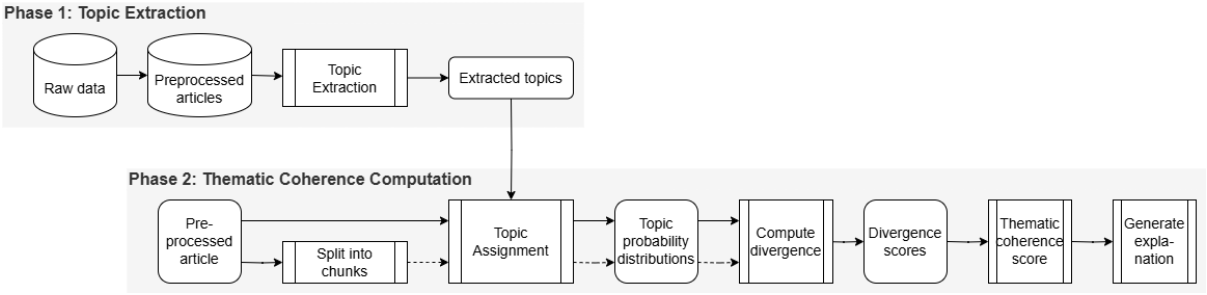


Figure 3.1: Thematic coherence computation procedure.

3.1 Phase one: Topic Extraction

The first phase of the proposed method is the topic extraction phase. It is designed to leverage topic modeling to extract topics from data. This phase comprises two key steps: pre-processing the dataset and extracting topics from it. How this is done, along with some key considerations, is outlined below.

3.1.1 Data and data pre-processing

Our approach does not pose many requirements on the type of input data used, even though we mainly focus on text data. Generally, more documents in the dataset mean a richer coverage of the latent - meaning underlying - topics present in that dataset. Also, a higher alignment between the dataset for topic extraction and the domain of target articles means a lower risk of non-insightful topics [91]. Therefore, it helps if the dataset is sufficiently large and aligned to the intended target domain. In our research, the datasets used are limited to news articles.

The extent of data pre-processing necessary depends on the dataset used for topic extraction. This research is limited to datasets containing news articles. In their raw form, news articles are generally quite well-suited for the procedure and require minimal modification. Nonetheless, if they are found to contain noise such as editor's notes, URLs, and source attributions, these should be removed. This ensures that only the actual news articles are used for topic extraction and prevents irrelevant information unrelated to the news content from influencing this process.

3.1.2 Topic extraction

The pre-processed articles are used as inputs for topic extraction, referring to the eponymous module in Figure 3.1. Topics are extracted from all the pre-processed articles using BERTopic. BERTopic is used because its transformer-based nature makes it the most suitable for analyzing shorter texts such as an article or its chunks. How BERTopic creates the topics is outlined in section 2.3.4. In this research, the following configuration is used for the different steps in the

algorithm.

The embedding model used is "all-MiniLM-L6-v2". This is a sentence transformer model that maps sentences and paragraphs to a 384-dimensional dense vector space. UMAP is selected for dimensionality reduction as it effectively captures both local and global patterns in high-dimensional data. For clustering, the density-based HDBSCAN algorithm is used, as it can identify clusters of varying shapes and densities while labeling outliers, improving the quality of the topic representation. Note that these outliers contribute to an "outlier probability," meaning the topic probability distribution may not sum to 1. HBDSCAN may generate clusters with different degrees of densities and varying shapes. Therefore, a topic representation technique is required that does not make assumptions about the expected structure of the clusters. To do this, all documents in a cluster are combined into a single document. That single - and very long - document represents that cluster. From here, the frequency of words in each cluster can be found on a cluster level. To account for clusters of different sizes, this bag-of-words representation is L1-normalized. At this point, we have a set of words that describe a collection of documents as generated by c-TF-IDF. To generate a topic representation, KeyBERTInspired is used, which leverages representative topic embeddings and calculates the semantic similarity between candidate keywords and the topic embedding using the same embedding model that embedded the documents. Although the generated representations are accurate, current developments in the NLP domain allow further fine-tuning of the representations. In this research, the external API from OpenAI (ChatGPT) is used to generate more interpretable topic labels. This results in a concise topic summary of a few wordsbover merely individual keywords which KeyBERTInspired would have outputted.

The output of this step is a trained topic model TM, which internally stores the discovered topics as a list. The number of topics may vary based on the size of the dataset and can potentially be reduced for better interpretability [79]. Once the topics are created, the first phase of the procedure is completed.

3.2 Phase two: Thematic Coherence Computation

The second phase involves computing a thematic coherence score for an individual article. Thematic coherence quantifies how consistently smaller sections of an article (referred to as chunks) align with its central theme(s). To calculate the thematic coherence, the topics of the entire article (representing its main theme(s)) are determined first. The article is then divided into chunks, and the topical divergence between each chunk and the main theme(s) is measured. These divergences are aggregated into a final thematic coherence score, which evaluates the extent to which the article remains focused on its main topics throughout.

3.2.1 Data and data pre-processing

The input text can originate from the same dataset used for topic extraction or from an unrelated source. In our case, the used texts are news articles. The pre-processing requirements are the same as for the extraction process. Moreover, the text must contain at least three sentences to ensure the chunking procedure works correctly. For illustrative purposes, a running example will be included at each step. The example will be modified accordingly to help visualize the functionality being discussed.

3.2.2 Split into chunks

After pre-processing an article, it is used as input for the 'split into chunks' module, as shown in Figure 3.1. To measure the thematic coherence within an article, the article needs to be split up into different segments called chunks. We intend to capture the thematic coherence over multiple ($\geq$ 3) chunks of an article. A chunk is defined as a contiguous segment of an article containing a number of consecutive sentences - this number can be parametrized and is set to five in this thesis. Three chunks are chosen as a minimum to provide a more comprehensive measure of thematic coherence than the existing literature, which often focuses solely on coherence between the title/opening sentences and the body of the text [70] [19]. The pseudocode for chunk creation is outlined in Algorithm 1. Before elaborating upon the specific steps in the algorithm, it is important to note that the main goal is to create at least three chunks to be able to measure the progression of topics over multiple chunks in the text. The second goal is to create uniform chunks of a desired chunk size. However, if the desired size cannot be reached, this desired chunk size is adjusted downwards to guarantee the creation of at least three chunks.

Algorithm 1 SPLITINTOCHUNKS

Require: text T, minimum_number_of_chunks *minChunks*, desired_sentences_per_chunk S

1: $sentences \leftarrow \text{PARSETEXT}(T)$
2: $n \leftarrow \text{LENGTH}(sentences)$
3: $actualNumChunks \leftarrow \lfloor n \,/\, S \rfloor$
4: **if** $actualNumChunks < minChunks$ **then**
5: $\quad actualNumChunks \leftarrow minChunks$
6: **end if**
7: $chunkSize \leftarrow \lfloor \frac{n}{actualNumChunks} \rfloor$
8: $leftoverSentences \leftarrow n \bmod actualNumChunks$
9: $chunkSizes \leftarrow [\, chunkSize \,] \times actualNumChunks$
10: **for** $i \leftarrow 0$ to $(leftoverSentences - 1)$ **do**
11: $\quad chunkSizes[i] \leftarrow chunkSizes[i] + 1$
12: **end for**
13: $chunks \leftarrow [\,]$
14: $startIdx \leftarrow 0$
15: **for each** *size* **in** *chunkSizes* **do**
16: $\quad newChunk \leftarrow \text{JOINSENTENCES}(sentences[startIdx : startIdx + size])$
17: $\quad \text{APPEND}(chunks, newChunk)$
18: $\quad startIdx \leftarrow startIdx + size$
19: **end for**
20: **return** *chunks*

Lines 1-6 are concerned with ensuring a sufficient number of chunks. After the input article is split into individual sentences (lines 1–2), the algorithm initially assumes each chunk should contain the desired number of sentences S and thus sets $actualNumChunks = \lfloor n/S \rfloor$ in line 3, where n is the total number of sentences. However, if this value is smaller than the required minimum (*minChunks*) number of chunks, the algorithm overrides it by setting $actualNumChunks = minChunks$ in line 5, ensuring at least *minChunks* total chunks will be formed.

After the number of chunks *actualNumChunks* is decided, in line 7-11 the algorithm creates chunk sizes as uniformly as possible. Line 7 computes a base chunk size, $chunkSize = \lfloor n/actualNumChunks \rfloor$. Line 8 determines the $leftoverSentences = n \bmod actualNumChunks$. Next, line 9 initializes *chunkSizes* as an array of length *actualNumChunks*, each entry set to *chunk_size*. Because integer division can leave leftover sentences, lines 10–12 add $+1$ to the first few chunks (up to the *leftoverSentences*) so that all sentences are accounted for. This way, all chunks differ in size by at most one sentence.

Finally, the algorithm assembles the actual text chunks in lines 13–19. It begins with an empty list *chunks* and a pointer *startIdx* at zero (lines 13–14). It then iterates through the entries of *chunkSizes*, each representing how many sentences belong in the next chunk. For each size, the algorithm slices out that many consecutive sentences from the original list (line 16) and appends the resulting text to *chunks*. At the end of the loop, line 20 returns the final list of

chunks.

For illustration purposes, consider a 25-sentence article ($n = 25$) with *minChunks* $= 3$ and $S = 5$. In line 3, $\lfloor 25/5 \rfloor = 5$ chunks; since $5 \geq$ *minChunks* $= 3$, no override is needed in line 4–5. Hence, lines 7–8 yield *chunkSize* $= \lfloor 25/5 \rfloor = 5$ and *leftoverSentences* $= 25 \bmod 5 = 0$. Lines 9–12 create *chunkSizes* $= [5, 5, 5, 5, 5]$. Finally, lines 13–20 build five contiguous chunks, each containing exactly five sentences.

Alternatively, if an article has 13 sentences, $\lfloor 13/5 \rfloor = 2$. Because $2 <$ *minChunks* $= 3$, line 5 sets *actualNumChunks* $= 3$. Then lines 7–8 compute *chunk_size* $= \lfloor 13/3 \rfloor = 4$ and *leftoverSentences* $= 13 \bmod 3 = 1$. Thus, line 9 initializes *chunkSizes* $= [4, 4, 4]$, and line 11 adds +1 to the first chunk, yielding $[5, 4, 4]$. The final chunks consist of a first chunk with five sentences, then two with four.

A six-sentence article is used as a running example. The full article and its corresponding chunks are outlined below in Figure 3.2.

Full text South Korean nuclear experts, checking for contamination after North Korea's sixth and largest nuclear test, said on Friday they have found minute traces of radioactive xenon gas but that it was too early to specify its source. The Nuclear Safety and Security Commission (NSSC) said it had been conducting tests on land, air and water samples since shortly after North Korea's nuclear test on Sunday. The statement said the commission was analyzing how the xenon entered South Korean territory and will make a decision at a later time whether the material is linked to North Korea's nuclear test . Xenon is a naturally occurring, colorless gas that is used in manufacturing of some sorts of lights. But the NSSC said it had detected xenon-133, a radioactive isotope that does not occur naturally and which has in the past been linked to North Korea's nuclear tests. There was no chance the xenon will have an impact on South Korea's territory or population , the statement said.
Chunk 1 South Korean nuclear experts, checking for contamination after North Korea's sixth and largest nuclear test, said on Friday they have found minute traces of radioactive xenon gas but that it was too early to specify its source. The Nuclear Safety and Security Commission (NSSC) said it had been conducting tests on land, air and water samples since shortly after North Korea's nuclear test on Sunday.
Chunk 2 The statement said the commission was analyzing how the xenon entered South Korean territory and will make a decision at a later time whether the material is linked to North Korea's nuclear test . Xenon is a naturally occurring, colorless gas that is used in manufacturing of some sorts of lights.
Chunk 3 But the NSSC said it had detected xenon-133, a radioactive isotope that does not occur naturally and which has in the past been linked to North Korea's nuclear tests. There was no chance the xenon will have an impact on South Korea's territory or population , the statement said.

Figure 3.2: Full input text and resulting chunks outputted by the 'split into chunks' module.

3.2.3 Topic assignment

The topic assignment module takes the trained topic model containing the extracted topics, which is the output of phase 1, as input. This topic assignment step uses these extracted topics and assigns them to both the full article and the chunks (solid and dashed line respectively in Figure 3.1). The purpose of this step is to compute topic probability distributions. A topic probability distribution shows how likely a text is to belong to each of the extracted topics. It highlights which topics are most prevalent in the text and how strongly they are represented. The assignment of topics to the full article and chunks is done using BERTopic's `transform` function, which calculates the semantic similarity between the input text and the topic clusters

learned during the topic extraction phase.

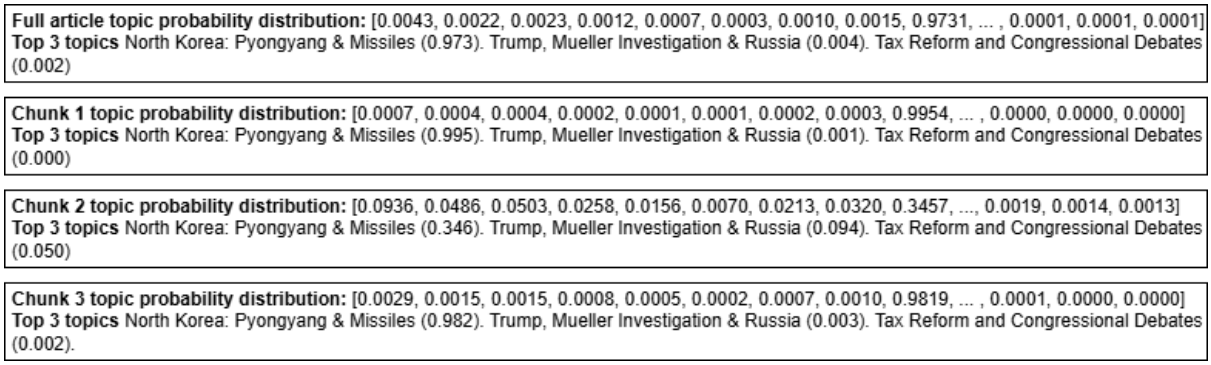


Figure 3.3: Topic probability distributions and top three topics for the full article and chunks (as seen in Figure 3.2).

Figure 3.3 shows the topic assignment step and shows the corresponding outputs, meaning the topic probability distributions for the full article and the chunks (as displayed in Figure 3.2). Note that the topic probability distributions are n-dimensional, where n is equal to the number of topics extracted in phase one. Truncated versions of the topic probability distributions are shown in the figure, as well as the three topics with the highest probabilities in the distribution and their corresponding, fine-tuned labels. The latter are included for illustrative purposes to make the raw topic probability distributions more interpretable.

3.2.4 Compute divergence

To assess how coherently topics are discussed within a text, the topic probability distribution of the full article is compared to those of the chunks. There are several methods to compare divergence between probability distributions. In our study, there is one aspect that is critical in determining a divergence measure: its ability to deal with sparse or missing probabilities. Given that we are analyzing smaller text chunks, they may entirely lack certain topics extracted from the full corpus. Occasionally, missing or sparse topics can occur, reflecting a chunk's limited scope. Therefore, a measure is required that can account for this sparsity and does not become undefined for zero probabilities. Consequently, in this study, we use the Jensen-Shannon (JS) divergence to compare two topic probability distributions. JS divergence is a symmetric version of the Kullback-Leibler (KL) divergence, defined as the average of the KL divergence of each distribution to the mean of the two distributions:

$$D_{\mathrm{JS}}(P_X \parallel P_Y) = \frac{1}{2} D_{\mathrm{KL}}(P_X \parallel M) + \frac{1}{2} D_{\mathrm{KL}}(P_Y \parallel M) \tag{3.1}$$

where $M = \frac{1}{2}(P_X + P_Y)$ is the mean distribution - representing the averaged probability values of P_X and P_Y, serving as a midpoint - and D_{KL} is the Kullback-Leibler divergence. The KL divergence between two distributions P and Q is defined as $D_{KL}(P \parallel Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)}$. The JS divergence provides a bounded, interpretable range between 0 and 1, where 0 indicates identical distributions. This allows us to easily assess how much two distributions deviate from each other, where low divergence values indicate, in our context, close alignment between the chunk and the full article in terms of topics.

In our research, we apply this general formula to compare the topic probability distribution of the full article (P_A) with that of each chunk (P_{C_i}), where i refers to the i-th chunk. Specifically, P_A represents the topic probability distribution of the full article, and P_{C_i} represents the topic probability distribution of chunk i. The divergence score $D_{\text{JS}}(P_A \parallel P_{C_i})$ quantifies the alignment between the full article and each chunk, with lower values indicating better alignment and thus greater thematic similarity. The output of this step is a list of k divergence scores, one for each chunk. To illustrate how the divergence calculation works, only the divergence calculation between the topic probability distribution of the full article and the first chunk is shown, but the same process is normally repeated for the other chunks. Table 3.1 shows the truncated versions of the topic probability distributions for the full article (P_A) and the first chunk (P_{C_1}). These distributions assume n topics. Then, the mean probability distribution M is calculated by taking the average of the values for (P_A) and (P_{C_1}). This is computed to calculate the final Jensen-Shannon divergence score.

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**...**	n
P_A	0.0043	0.0022	0.0023	0.0012	0.0007	0.0003	0.0010	0.0015	0.9731	...	0.0001
P_{C_1}	0.0007	0.0004	0.0004	0.0002	0.0001	0.0001	0.0002	0.0003	0.9954	...	0.0000
M	0.0025	0.0013	0.0014	0.0007	0.0004	0.0002	0.0006	0.0009	0.9843	...	0.0001

Table 3.1: Divergence score for the topic probability distributions of the full article (P_A) and the first chunk (P_{C_1}).

The divergence score $D_{\text{JS}}(P_A \parallel P_{C_1})$ is then calculated as follows:

$$D_{\text{JS}}(P_A \parallel P_{C_1}) = 0.5D_{\text{KL}}(P_A \parallel M) + 0.5D_{\text{KL}}(P_{C_1} \parallel M) = 0.026 + 0.041 = 0.067$$

The table already signals that the topic probabilities are highly aligned, with both the full article and the chunk primarily discussing topic 9. The low final divergence score of 0.067 supports this, meaning the first chunk is thematically very similar to the full article. Repeating this process for the second and third chunk gives $D_{\text{JS}}(P_A \parallel P_{C_2})$ = 0.492 and $D_{\text{JS}}(P_A \parallel P_{C_3})$

= 0.021. This means that the second chunk is moderately aligned with the full article topics, while the third chunk is very similar to the full article in terms of topics. The final list of divergence values for our running example, which will also be used in the next step, becomes $[0.067, 0.492, 0.021]$.

3.2.5 Thematic coherence score

The final step in the computation of the thematic coherence score involves aggregating the values in the list of k JS divergence scores obtained in the previous step. Below, we discuss possible metrics for this aggregation (including their considerations) that one could use for this step.

- *Mean divergence.* The first metric is the mean divergence $\overline{D_{\text{JS}}} = \frac{1}{k}\sum_{i=1}^{k} D_{\text{JS}}(P_A \parallel P_{C_i})$. Here k is the number of chunks. This simply measures the average of the JS divergence values. This measure gives an overview of the extent to which the chunk topics overlap with the full article topic. However, it fails to capture variations between chunks as you lose information by aggregating. To illustrate: if the JS values consistently alternate between 0 and 1, averaging the JS values would indicate moderate dissimilarity, while in fact, the values show high fluctuations and thus may be indicative of an incoherent article.
- *Standard deviation.* In order to measure these fluctuations, the standard deviation may give insights. Standard deviation, in this context, is defined as: $\sqrt{\frac{1}{k}\sum_{i=1}^{k}\left(D_{\text{JS}}(P_A \parallel P_{C_i}) - \overline{D_{\text{JS}}}\right)^2}$. Here, $\overline{D_{\text{JS}}}$ is the mean JS divergence across all chunks as calculated above. A high standard deviation indicates large variations between chunks, suggesting the article discusses themes more inconsistently across different segments. A drawback of considering standard deviation is that it does not indicate any directionality or spread patterns - this could be important to understand where the incoherence occurs, giving insight into the topic dissimilarity between adjacent chunks.
- *Oscillation.* To take this spread into account, a measure of oscillation is needed. This can be measured by the first-order differences: $\frac{1}{k-1}\sum_{i=1}^{k-1}\left|D_{JS}(P_A \parallel P_{C_{i+1}}) - D_{JS}(P_A \parallel P_{C_i})\right|$. This captures how much JS divergence values change from one chunk to the next. A high oscillation indicates significant changes in topic dissimilarity, meaning topics progress incoherently.
- *Peaks.* Sudden shifts between topics within an article will be captured by analyzing peaks. A spike in the dissimilarity between the topic distributions indicates an unrelated topic is suddenly being introduced. This can be captured through a simple peak analysis. Peak analysis has been proven useful to detect sudden changes in other domains, like in the medical world, for monitoring EEG signals [92]. Although the appli-

cation is different, analyzing peaks in the fake news domain can indicate a sharp shift in topics reflecting reduced coherence. Therefore, the number of peaks and the peak ratio (the relative proportion of chunks where peaks are observed) may be computed. A peak is defined as a value in the sequence of JS values that is greater than its immediate neighboring values by a threshold (set to 0.15). More peaks suggest more frequent and random topic shifts and thus lower coherence, whereas a high peak ratio highlights that such topic shifts occur more consistently across an entire text.

- *Root Mean Square Error (RMSE).* The RMSE is a measure that captures the overall magnitude of divergence between chunk topics and the full article topic by taking both the mean and the variability into account. In this research, is defined as:

$$\text{RMSE} = \sqrt{\frac{1}{k}\sum_{i=1}^{k}\left(D_{\text{JS}}(P_A \parallel P_{C_i})\right)^2} \tag{3.2}$$

  Here, k is the number of chunks, and $D_{\text{JS}}(P_A \parallel P_{C_i})$ is the JS divergence between the topic probability distribution of the full article and chunk i. Unlike the mean divergence, which averages the values, or the standard deviation, which measures variability around the mean, the RMSE, to some extent, captures both aspects: magnitude through averaging and spread through squaring the divergence values.

Different metrics are possible, depending on the requirements of the research. It is important to note that in the research experiments outlined in chapter 5, the RMSE is used as it captures both the magnitude of divergence and the variability across chunks, addressing some key aspects of the metrics discussed above. While it does not encompass all dimensions—such as sudden shifts or fluctuations captured by peaks or oscillation—it provides the most balanced single measure for assessing thematic coherence. Filling in the obtained JS values from the previous step $[0.067, 0.492, 0.021]$ in equation 3.2 gives us the RMSE value used as the final thematic coherence score. This is outlined below, thereby completing the running example:

$$\text{RMSE} = \sqrt{\frac{1}{3}\left((0.067)^2 + (0.492)^2 + (0.021)^2\right)} = 0.287$$

This score tells us that, on average, the chunks are quite closely aligned with the full article topics, reflecting reasonably strong thematic coherence. Alternatively, a score of e.g. 0.7 would have indicated a substantially higher deviation between the topic probability distribution of

the full article and the chunks, meaning notably lower thematic coherence.

The final step related to obtaining the thematic coherence score is performed by calculating the score. Algorithm 2 below concisely recapitulates each step in the thematic coherence computation process and clearly states its inputs and outputs. It takes in a pre-processed text T, the trained topic model TM (as outlined in section 3.1), a minimum number of chunks *minChunks* and a desired number of sentences per chunk S. In line 1 it computes the chunks (by running Algorithm 1). In line 2, it generates the topic probability distributions (TPD) using BERTopic's TRANSFORM function on both the full article and the chunks. These are used to compute the JS divergence in line 3, followed by the coherence score by means of the RMSE calculation based on the JS values (line 4). Finally, it outputs a thematic coherence score. This score will be used in the future experiments.

Algorithm 2 COMPUTETHEMATICCOHERENCE

Require: text T, topic_model TM, minimum_number_of_chunks *minChunks*, desired_sentences_per_chunk S

1: $chunks \leftarrow$ SPLITINTOCHUNKS$(T,\ minChunks,\ S)$
2: $fullArticleTPD,\ chunkTPDs \leftarrow$ ASSIGNTOPICS$(TM,\ T,\ chunks)$
3: $jsValues \leftarrow$ COMPUTEJS$(fullArticleTPD,\ chunkTPDs)$
4: $coherenceScore \leftarrow$ COMPUTECOHERENCESCORE$(jsValues)$
5: **return** *coherenceScore*

3.2.6 Generate explanation

We have now obtained a thematic coherence score. Several examples have been included at each step in the proposed method to showcase its functionality. However, given the importance of explanations for AI models, we intend to incorporate the complete method into a final algorithm that also generates a comprehensive explanation. The motivation for developing and integrating such explanations stems from the significant role they play in bridging the gap between model behavior and user understanding [93]. Specifically, we propose an explanation approach meant to describe our model's decision-making process.

The explanation incorporates all elements of the thematic coherence computation procedure as outlined in section 3.2. It begins by splitting the article of interest into chunks. Next, topics are assigned, and the top three most probable topics are displayed alongside the text. For each topic, the extracted keywords that best represent the topic based on their c-TF-IDF scores are shown, highlighting the most relevant words for that specific topic. Each topic is color-coded, and keywords belonging to that topic that occur in the text are marked in their respective color to gain an overview of the words contributing to the topic assignment. If a word is a keyword for e.g. two topics, it is marked horizontally in both colors. Using the topic probability distributions, JS values are calculated for each chunk. These are used to compute

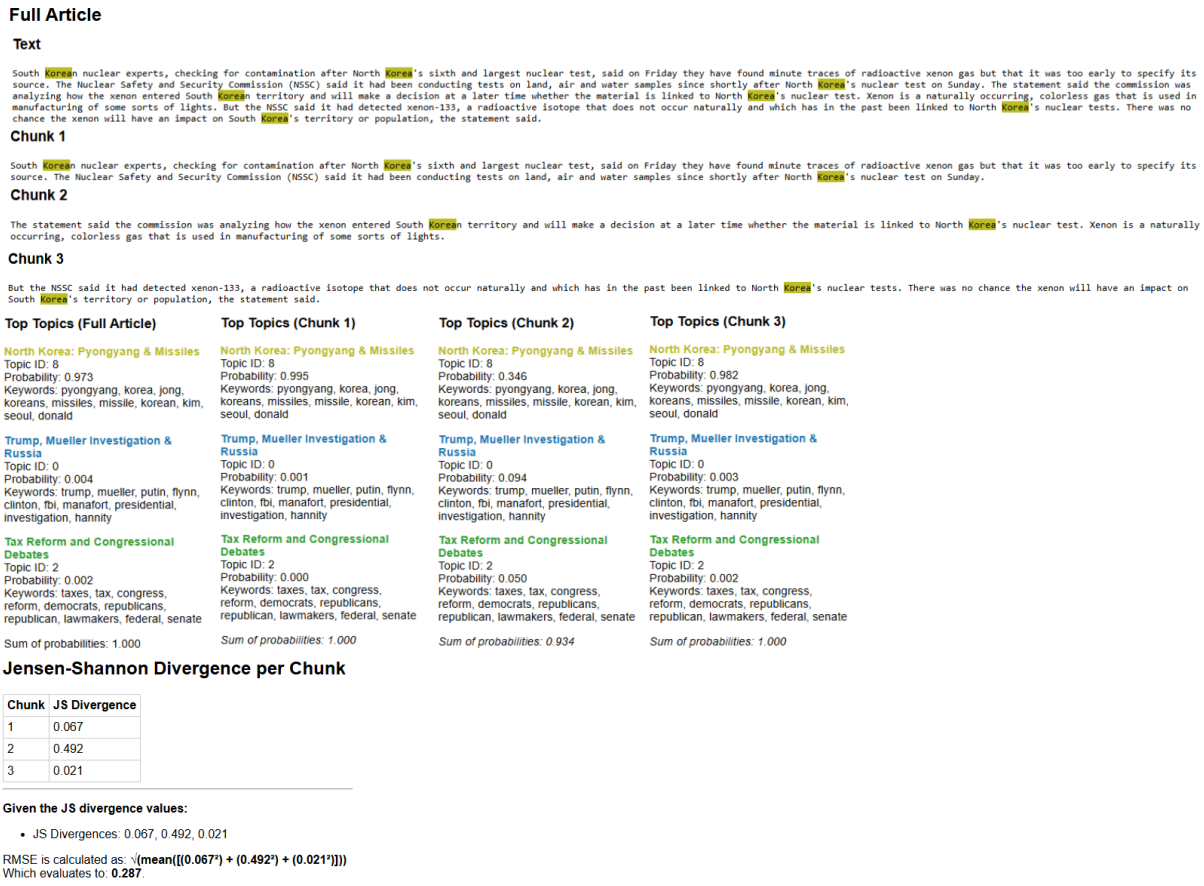
the final score, reflecting the overall thematic coherence of the text. The visualization of the explanation for the article used as the running example throughout this chapter (as displayed in Figure 3.2) is shown at the end of this section. The evaluation of a different example, along with some qualitative insights derived from the explanations, will be presented in section 5.4. This process justifies the model's decisions in an interpretable and visual manner, aiming to enhance users' trust and understanding. The pseudocode for generating explanations is outlined in Algorithm 3. Algorithm 3 extends Algorithm 2 with elements that support the generation of explanations.

Algorithm 3 GENERATEEXPLANATION

Require: text T, topic_model TM, minimum_number_of_chunks *minChunks*, desired_sentences_per_chunk S

1: $chunks \leftarrow$ SPLITINTOCHUNKS(T, $minChunks$, S)
2: $fullArticleTPD$, $chunkTPDs \leftarrow$ ASSIGNTOPICS(TM, T, $chunks$)
3: $fullTop3$, $chunkTop3 \leftarrow$ EXTRACTTOPTOPICS($fullArticleTPD$, $chunkTPDs$)
4: HIGHLIGHTKEYWORDS(T, $fullTop3$, $chunkTop3$)
5: $jsValues \leftarrow$ COMPUTEJS($fullArticleTPD$, $chunkTPDs$)
6: $coherenceScore \leftarrow$ COMPUTECOHERENCESCORE($jsValues$)
7: BUILDEXPLANATIONREPORT(T, $chunks$, $fullArticleTPD$, $chunkTPDs$, $fullTop3$, $chunkTop3$, $jsValues$, $coherenceScore$)

In lines 1–2, we begin by taking an article T from the dataset and splitting it into chunks using SPLITINTOCHUNKS as before. We then assign topics to both the full article and each chunk (line 2), retrieving the full-article topic probability distribution $fullArticleTPD$ and the chunk-level topic distributions *chunkTPDs*. Next, in line 3, we extract the top three topics $fullTop3$ and *chunkTop3* for the full article and each chunk, by selecting the three most probable topics from their respective topic probability distributions. In line 4, HIGHLIGHTKEYWORDS uses those top topics to identify and highlight the most relevant words within the original text T. This visually illustrates which words contributed most to the topic assignment in each segment. Line 5 COMPUTEJS measures how much each chunk's topic distribution deviates from the full article's distribution. Line 6 aggregates those divergence values via COMPUTECOHERENCESCORE, producing our thematic coherence score. Finally, in line 7, BUILDEXPLANATIONREPORT assembles a concise explanation that includes the text, the chunks, top topics, color-highlighted keywords, and the final thematic coherence score. An example explanation for the running example used throughout this chapter is illustrated in Figure 3.4 below. This explanation clarifies how the score is derived and offers insights into the article's topical structure.



Full Article

Text

South Korean nuclear experts, checking for contamination after North Korea's sixth and largest nuclear test, said on Friday they have found minute traces of radioactive xenon gas but that it was too early to specify its source. The Nuclear Safety and Security Commission (NSSC) said it had been conducting tests on land, air and water samples since shortly after North Korea's nuclear test on Sunday. The statement said the commission was analyzing how the xenon entered South Korean territory and will make a decision at a later time whether the material is linked to North Korea's nuclear test. Xenon is a naturally occurring, colorless gas that is used in manufacturing of some sorts of lights. But the NSSC said it had detected xenon-133, a radioactive isotope that does not occur naturally and which has in the past been linked to North Korea's nuclear tests. There was no chance the xenon will have an impact on South Korea's territory or population, the statement said.

Chunk 1

South Korean nuclear experts, checking for contamination after North Korea's sixth and largest nuclear test, said on Friday they have found minute traces of radioactive xenon gas but that it was too early to specify its source. The Nuclear Safety and Security Commission (NSSC) said it had been conducting tests on land, air and water samples since shortly after North Korea's nuclear test on Sunday.

Chunk 2

The statement said the commission was analyzing how the xenon entered South Korean territory and will make a decision at a later time whether the material is linked to North Korea's nuclear test. Xenon is a naturally occurring, colorless gas that is used in manufacturing of some sorts of lights.

Chunk 3

But the NSSC said it had detected xenon-133, a radioactive isotope that does not occur naturally and which has in the past been linked to North Korea's nuclear tests. There was no chance the xenon will have an impact on South Korea's territory or population, the statement said.

Top Topics (Full Article)

North Korea: Pyongyang & Missiles
Topic ID: 8
Probability: 0.973
Keywords: pyongyang, korea, jong, koreans, missiles, missile, korean, kim, seoul, donald

Trump, Mueller Investigation & Russia
Topic ID: 0
Probability: 0.004
Keywords: trump, mueller, putin, flynn, clinton, fbi, manafort, presidential, investigation, hannity

Tax Reform and Congressional Debates
Topic ID: 2
Probability: 0.002
Keywords: taxes, tax, congress, reform, democrats, republicans, republican, lawmakers, federal, senate

Sum of probabilities: 1.000

Top Topics (Chunk 1)

North Korea: Pyongyang & Missiles
Topic ID: 8
Probability: 0.995
Keywords: pyongyang, korea, jong, koreans, missiles, missile, korean, kim, seoul, donald

Trump, Mueller Investigation & Russia
Topic ID: 0
Probability: 0.001
Keywords: trump, mueller, putin, flynn, clinton, fbi, manafort, presidential, investigation, hannity

Tax Reform and Congressional Debates
Topic ID: 2
Probability: 0.000
Keywords: taxes, tax, congress, reform, democrats, republicans, republican, lawmakers, federal, senate

Sum of probabilities: 1.000

Top Topics (Chunk 2)

North Korea: Pyongyang & Missiles
Topic ID: 8
Probability: 0.346
Keywords: pyongyang, korea, jong, koreans, missiles, missile, korean, kim, seoul, donald

Trump, Mueller Investigation & Russia
Topic ID: 0
Probability: 0.094
Keywords: trump, mueller, putin, flynn, clinton, fbi, manafort, presidential, investigation, hannity

Tax Reform and Congressional Debates
Topic ID: 2
Probability: 0.050
Keywords: taxes, tax, congress, reform, democrats, republicans, republican, lawmakers, federal, senate

Sum of probabilities: 0.934

Top Topics (Chunk 3)

North Korea: Pyongyang & Missiles
Topic ID: 8
Probability: 0.982
Keywords: pyongyang, korea, jong, koreans, missiles, missile, korean, kim, seoul, donald

Trump, Mueller Investigation & Russia
Topic ID: 0
Probability: 0.003
Keywords: trump, mueller, putin, flynn, clinton, fbi, manafort, presidential, investigation, hannity

Tax Reform and Congressional Debates
Topic ID: 2
Probability: 0.002
Keywords: taxes, tax, congress, reform, democrats, republicans, republican, lawmakers, federal, senate

Sum of probabilities: 1.000

Jensen-Shannon Divergence per Chunk

Chunk	JS Divergence
1	0.067
2	0.492
3	0.021

Given the JS divergence values:

- JS Divergences: 0.067, 0.492, 0.021

RMSE is calculated as: $\sqrt{\textbf{(mean([(0.067}^2\textbf{) + (0.492}^2\textbf{) + (0.021}^2\textbf{)]))}}$
Which evaluates to: **0.287**.

Figure 3.4: Example of a generated explanation.

4. The Sentence Replacement Task: A Task for Evaluating Thematic Coherence

To assess whether coherence models can effectively capture text coherence, an evaluation task is needed. A coherence evaluation task tests the ability of a coherence model to distinguish coherent from incoherent texts. Traditionally, coherence models are evaluated on the sentence ordering task [29] [30] [84]. However, this task is less suited for evaluating higher-level thematic coherence, as it primarily focuses on sentence-to-sentence relationships. Therefore, we propose a new evaluation task that focuses specifically on thematic coherence. The following sections outline the sentence ordering task, its limitations, and the motivation for developing a more suitable approach to evaluating thematic coherence.

4.1 Sentence Ordering Task

In the sentence ordering task, a document is compared to a random permutation of its sentences. The goal of this task is to assess the ability of a coherence model to rank the original document higher in terms of coherence than the permuted one. The idea is that by scrambling the order of sentences, local patterns - such as entity transitions - are disrupted, causing the model to assign lower coherence scores to the permuted article. However, this approach overlooks the structure of a text that that contributes to its thematic coherence. For example, some sentences may be thematically coherent yet lack clear syntactic links (e.g. in the form of shared entities). As a result, the sentence ordering task may not succeed at evaluating the coherence at the thematic level. To illustrate this, let us consider the same example used in chapter 3. The top three topics from the original article are extracted, as well as the top three topics of its permuted version and they are shown in Figure 4.1.

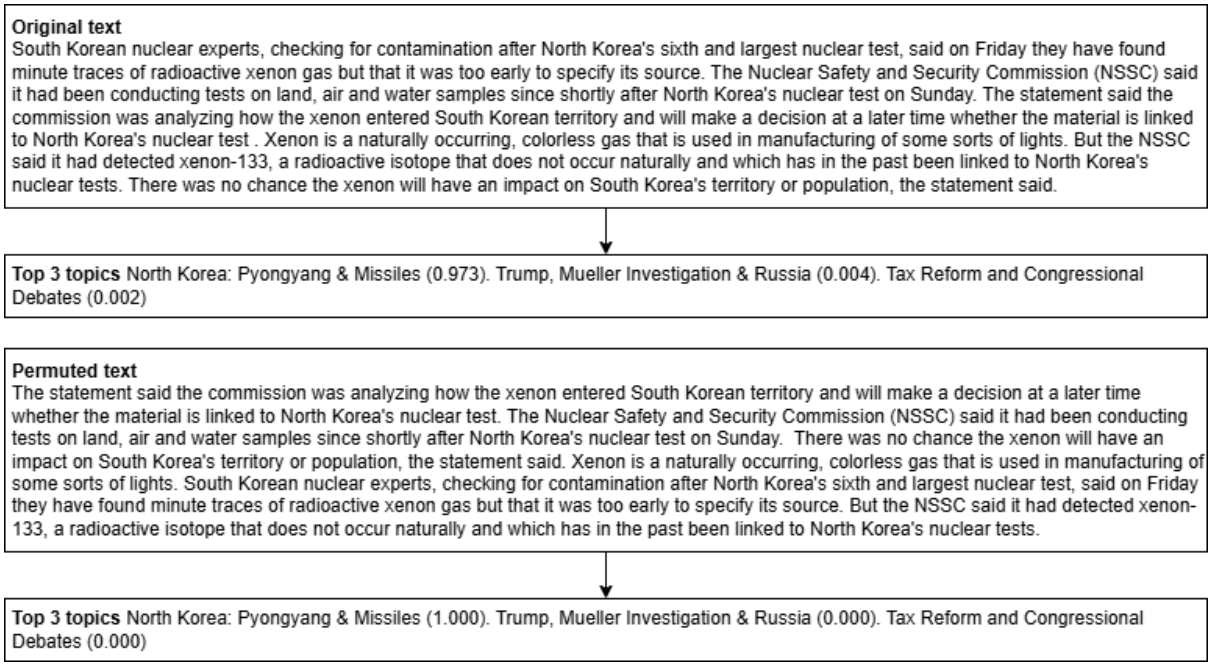


Figure 4.1: Top topics extracted from the original article and its permutation.

Extracting the top three topics reveals that scrambling the sentence order barely affects thematic coherence, and the topic probability distributions are still very similar. This highlights that reordering sentences does not necessarily introduce thematic incoherence. This might raise the question: Why use a thematic coherence measure if it cannot detect a scrambled article? To show local and global coherence can operate on distinct levels, we will present a scenario where a high performance on the sentence ordering task can mask thematic incoherence, providing additional motivation for a new evaluation task.

To illustrate this, let us consider an article that is locally coherent but thematically incoherent:

"President Biden addressed climate change, emphasizing renewable energy and environmental protection while advocating stronger international cooperation. LeBron James showcased his skills during a high-stakes basketball game, demonstrating strategic plays and leadership on the court. President Biden unveiled a healthcare reform plan, highlighting benefits for middle-class families and stressing the need for bipartisan support. LeBron James engaged in community outreach, visiting schools and organizing youth programs to promote education and social responsibility. President Biden discussed economic growth strategies, citing job creation and infrastructure improvements as key drivers of future prosperity. LeBron James participated in philanthropic events, donating to charities and supporting underprivileged communities."

This article maintains good local coherence due to the repeated entities and predictable transitions of those entities across sentences. However, its themes shift drastically — from sports to political topics such as the environment, healthcare, and the economy. When cre-

ating 20 permutations of the article and comparing them to the original (in accordance with prior work, as will be further discussed in section 5.1), the sentence ordering task obtains good accuracy of 85%, meaning the original is ranked better in terms of local coherence than its permutations 17 out of 20 times. However, upon reading the original article, its thematic incoherence becomes evident as it abruptly shifts between various unrelated topics. This illustrates that an article may be locally coherent and perform well on the sentence ordering task, but that this does not necessarily imply thematic coherence. In such cases, relying solely on sentence ordering may be insufficient to evaluate coherence comprehensively. To address this, we introduce a new sentence replacement task designed to directly assess thematic coherence.

4.2 Sentence Replacement Task

We have demonstrated that the sentence ordering task does not always adequately capture thematic coherence as it focuses primarily on local coherence transitions, which is different than what we intend to evaluate. To address this, the sentence replacement task is designed. The goal of this task is to assess the ability of a coherence model to distinguish thematically coherent from thematically incoherent texts. To assess this ability, we first need both coherent and incoherent texts. Similar to the sentence ordering task and in line with previous research [29] [30], it is assumed that the articles in their original form are thematically coherent. However, the previous section demonstrated that altering sentence order does not sufficiently disrupt thematic coherence in articles. To address this, we introduce a new approach to creating noise: replacing sentences. In this context, noise refers to intentionally disrupting thematic coherence by substituting sentences within the article (instead of scrambling the sentence order done previously). The pseudocode for this replacement task is shown in Algorithm 4 and will be further elaborated upon below.

Algorithm 4 SENTENCEREPLACEMENTTASK

Require: text T, topicModel TM, minimum_number_of_chunks *minChunks*, desired_sentences_per_chunk S, sentence_pool SP, percentage_chunks_affected CA, num_sentences_replaced_per_chunk SR, num_trials t

1: $originalCoherenceScore \leftarrow$ COMPUTETHEMATICCOHERENCE$(T,\ TM,\ minChunks,\ S)$
2: correctCount $\leftarrow 0$
3: totalVariants $\leftarrow 0$
4: **for** trial $\leftarrow 1$ **to** t **do**
5: $\quad chunks \leftarrow$ SPLITINTOCHUNKS$(T,\ minChunks,\ S)$
6: $\quad x \leftarrow CA \times$ LENGTH$(chunks)$
7: $\quad$ **if** $0 < x < 1$ **then**
8: $\quad\quad \ell \leftarrow 1$
9: $\quad$ **else**
10: $\quad\quad \ell \leftarrow$ ROUND(x)
11: $\quad$ **end if**
12: $\quad modifiedTexts \leftarrow$ CREATEMODIFIEDTEXTS$(T,\ TM,\ chunks,\ SP,\ \ell,\ SR)$
13: $\quad$ **for each** m **in** $modifiedTexts$ **do**
14: $\quad\quad modifiedCoherenceScore \leftarrow$ COMPUTETHEMATICCOHERENCE$(m,\ TM,\ minChunks,\ S)$
15: $\quad\quad$ **if** $originalCoherenceScore < modifiedCoherenceScore$ **then**
16: $\quad\quad\quad$ correctCount $\leftarrow$ correctCount $+ 1$
17: $\quad\quad$ **end if**
18: $\quad\quad$ totalVariants $\leftarrow$ totalVariants $+ 1$
19: $\quad$ **end for**
20: **end for**
21: accuracy $\leftarrow \dfrac{\text{correctCount}}{\text{totalVariants}}$
22: **return** STORERESULTS(accuracy)

First, let us consider the inputs for this algorithm. T is the original article text, and TM is the trained topic model containing the extracted topics from phase one (in section 3.1). The percentage of chunks to affect (CA), the number of sentences to replace per chunk (SR), and the number of trials (t) are all parameters. Because our method contains some randomness in generating modifications, each trial repeats the same process. This mitigates the variability introduced by this randomness. The number of trials in this research is set to three. The sentence pool (SP) is created beforehand and consists of a dataset of sentences along with their corresponding topic probability distributions. To better understand what happens when applying the sentence replacement task to a single article under given configurations (CA) and (SR), a more detailed explanation is given below.

Lines 1–3 of Algorithm 4 handle the initial setup. Line 1 calls COMPUTETHEMATICCOHERENCE to obtain the coherence score for the input text; lines 2–3 initialize counters for the number of correct comparisons and total variants. Note that we intentionally opt for the name 'variants' to avoid confusion with 'permutations' used earlier for the sentence ordering task. 'Variants' refers to the number of modified texts generated, not to the individual replacement operations. Lines 4–20 form the outer loop over the number of trials (t). Within each trial,

line 5 splits the article T into *chunks* via SPLITINTOCHUNKS, thereby creating chunks. Line 6 uses this to calculate the actual number of chunks that will be affected. In lines 7-11, we check this number: if $0 < x < 1$, we set $\ell = 1$; otherwise, we round x to the nearest integer. This ensures we never end up replacing fewer than one chunk nor exceeding the intended fraction of chunks.

Once ℓ is determined, line 12 creates *modifiedTexts* using CREATEMODIFIEDTEXTS. This function contains quite some functionality: first, it extracts the top topic from the original text T using TM. It then randomly selects a number (ℓ) of chunks from the previously created *chunks* for modifications. In each selected chunk, a number of sentences (SR) are replaced by sentences drawn from the sentence pool (SP). Replacement sentences are chosen such that their top topic is dissimilar to the top topic of T, ensuring that the modifications introduce thematic incoherence. There is still a chance that a similar sentence is being used for replacement if the top topics of the two sentences under analysis happen to be dissimilar but the remainder of the topic probability distribution is aligned. This risk is also mitigated by running multiple trials. Putting this together, this function creates a number of modified texts *modifiedTexts* equal to the number of chunks present in the original text. Setting the number of modified texts created equal to the number of chunks (over e.g. a fixed amount), ensures that modifications scale with article length. This prevents redundant modifications in shorter articles, making evaluation consistent regardless of article length.

Lines 13–19 proceed by computing the thematic scores for the modified texts *modifiedCoherenceScore*, comparing them to the score of the original text *originalCoherenceScore*. If the original text is thematically more coherent (*originalCoherenceScore* < *modifiedCoherenceScore*), we increment correctCount. Line 21 computes the accuracy as the proportion of comparisons where the original article is correctly classified as thematically more coherent. An accuracy of 100% indicates that the original article was always classified as more coherent, whereas lower accuracies suggest cases where the modifications were misclassified as more coherent than the original. The results are finally returned via STORERESULTS.

To illustrate the workings of this evaluation task, let us consider the same original text used in the example shown in Figure 4.1. Note that this highlights only one trial with CA=50% and SR=1.

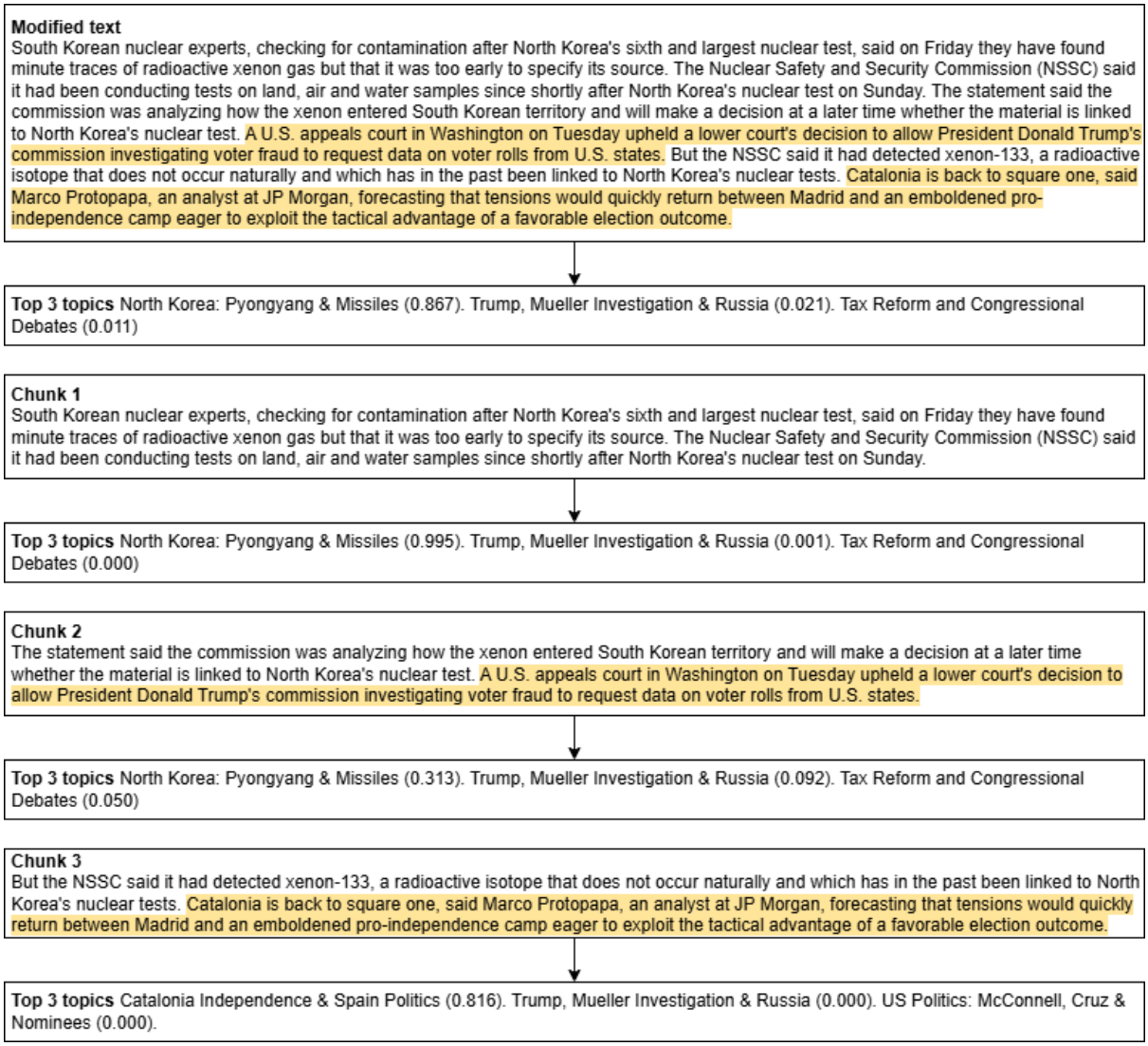


Figure 4.2: Top topics extracted for the article and its chunks after sentence replacement.

Walking through the steps in Algorithm 4, we first compute the *originalCoherenceScore* (this was also done in section 3.2.5 and evaluated to 0.287). The function SPLITINTOCHUNKS splits the six-sentence article into three chunks containing two sentences each. We proceed by determining the amount of chunks to be affected. In line 6, we obtain 1.5 chunks to be affected (50% of three). Based on the rounding in line 10, this leads us to modify two chunks. These two chunks are chosen randomly (chunk two and three in Figure 4.2). One unrelated sentence is selected for replacement for both chunks as SR=1. After determining the chunks to be affected and the sentences to be replaced, the modified text *modifiedText* is constructed using the function CREATEMODIFIEDTEXTS. The modified text can be seen at the top of Figure 4.2, with the replaced sentences highlighted in yellow. We then calculate the *modifiedCoherenceScore* calling Algorithm 2 for the modified text. The corresponding chunks are also shown in the Figure 4.2. After computing the divergence, we obtain the list of modified divergence values $[0.200, 0.388, 0.826]$. This results in a *modifiedCoherenceScore* = 0.539. This means that

$originalCoherenceScore < modifiedCoherenceScore = 0.287 < 0.539$ is true and the model has correctly classified the original as thematically more coherent.

Furthermore, it is interesting to observe what happens with the top three topics for the chunks containing the replaced sentences (which are highlighted for visualization purposes only). The first chunk is unaffected, so the top three topics are identical to the ones in Figure 3.3 and we also see that for the full article, the most prevalent topic is still 'North Korea: Pyongyang and Missiles' (albeit slightly less probable). For the second chunk, the inclusion of a replaced sentence minimally alters the original topic probability distribution. In contrast, the third chunk shows that the replaced sentence tremendously impacts the extracted topic probability distribution: now, 'Catalonia Independence & Spain Politics' is the main topic, which is very unaligned with the full article topics. This is reflected in the divergence value for the third chunk (0.826), which significantly inflates the overall incoherence of the modified article (0.539).

5. Evaluation

This chapter will describe how the proposed method is instantiated, outline the conducted experiments, and present the obtained results. The exact implementation (including e.g. parameter settings and required libraries) can be found in the Github repository[1] of this project. The experiments aim to test the proposed method in several ways. First, in section 5.1, its ability to capture thematic coherence will be investigated, seeking to answer the first research question (RQ1). This will also test the method on the newly proposed task, pertaining to the second research question (RQ2). After that, in section 5.2 it will be tested to what extent the proposed method can distinguish real from fake news articles, linking back to the third research question (RQ3). Third, in section 5.3, its ability to recognize LLM-generated text will be analyzed to answer the fourth and final research question (RQ4). Finally, some qualitative insights obtained from the explanations are discussed in section 5.4. For sections 5.1, 5.2 and 5.3, the structure is as follows. First, the experimental research question is outlined. Then, the dataset used and pre-processing steps taken to conduct the experiment are described. Afterwards, the instantiation of the method adopted to answer the experimental research question is given and the hypothesis is proposed. Finally, the results of the experiment are presented, and a discussion of the results is given, which reflects on the hypothesis.

5.1 Coherence evaluation tasks

The first question our experiments aim to provide an empirical answer to links back to research question RQ1 and is concerned with what method can be used to effectively capture thematic coherence in news articles. In order to answer this question, the proposed method outlined in chapter 3 is tested and the following experimental research question has been devised.

ERQ1 To what extent can the proposed thematic coherence method accurately distinguish between coherent and incoherent texts?

5.1.1 Dataset

Following prior work [86] [84], we use the Wall Street Journal (WSJ) portion of the Penn Treebank for both coherence evaluation tasks. It is assumed that these articles in their original form are coherent [29] [84]. The basic statistics on the WSJ dataset can be seen below in Table

[1] https://github.com/Willems-source/MScThesis

5.1. Some previous studies have used the 'airplanes' or 'earthquakes' corpora, which contain reports on airplane crashes and earthquakes, for which the average number of sentences per article is 10.4 and 11.5 respectively [29] [71]. The WSJ dataset is preferred in this research due to its larger average sentence length, which is helpful in assigning more meaningful topics, and its informative style, which more closely resembles real-life news articles.

Number of articles	**Avg. sentences/article**
1836	33.17

Table 5.1: Summary statistics for the WSJ dataset.

5.1.2 Evaluation task 1: Sentence ordering

The first experiment conducted is the sentence ordering task. We obtain the scores used for ranking the original versus the permuted articles by running Algorithm 2 for both the original and the permutations. In accordance with previous work, we will test 20 permutations for each document. Permutations that match the original article text are excluded [84] [27]. Using BERTopic's default HBDSCAN parameters, we extract 24 different topics from the WSJ dataset. All extracted topics are shown in the appendix (Table 1). These topics are then assigned to the full article and the chunks of both the original and the permuted article. The accuracy (defined below in equation 5.1) determines the ratio of comparisons where the original article's thematic coherence score is better (lower) than the permuted article's coherence score, relative to the total number of comparisons:

$$Accuracy = \frac{Number\ of\ Correct\ Comparisons}{Total\ Number\ of\ Comparisons} \tag{5.1}$$

Given that there's randomness introduced by randomly scrambling sentences, we conduct multiple trials (three in total). This leads to $1836 \times 3 \times 20 = 110{,}160$ comparisons overall (where 1836 is the number of articles, 3 is the number of trials, and 20 is the number of permutations per article).

5.1.3 Evaluation task 2: Sentence replacement

The second evaluation task is the sentence replacement task. As shown in Algorithm 4, we modify articles based on the percentage of chunks affected and the number of sentences replaced per chunk. The values for these parameters may vary, and we have explored the following values:

- Percentage of chunks affected: 10%, 20%, 30%, 50%, 75% and 100%.
- Number of sentences replaced per chunk: 1, 2, 3, 4 and all.

The values above require some elaboration. First of all, we chose for a percentage for the chunks affected, as the number of chunks may vary depending on the length of the article being analyzed. Moreover, we assume that it is more realistic in real life that minor disruptions are made. To reflect this, we've taken smaller steps initially, and as the percentages of chunks affected increase, we take bigger steps. For the number of sentences replaced, we do use absolute values, as the number of sentences per chunk is fixed at five sentences per chunk.[2]

Given that we have six different values for the percentage of chunks affected, five for the number of sentences replaced per chunk and we perform three trials per configuration, a total 90 runs are conducted. The goal of the experiment is to test the model's ability to accurately distinguish between thematically coherent articles (the original) and incoherent articles (the modified ones). The original coherence score is the output of Algorithm 2 and accuracies in ranking the scores of the original versus the modified articles are outputted by Algorithm 4. These accuracies are computed for each configuration, in a similar fashion as in the sentence ordering task before (using equation 5.1). Varying the degree of disruption in such a controlled manner should give us insight into the model's ability to detect thematically incoherent articles, as well as its sensitivity to different configurations.

Following the considerations outlined in chapter 4, we construct two hypotheses to help us determine if the proposed method can accurately distinguish between coherent and incoherent texts:

H1.1 The proposed method achieves an accuracy equal to random guessing on the sentence ordering task.

H1.2 The proposed method achieves an accuracy equal to random guessing on the sentence replacement task.

5.1.4 Sentence ordering results

The sentence ordering task was conducted to assess the model's ability to recognize the original text from its permuted counterparts. The results are presented in Table 5.2. Besides our primary measure, which is the RMSE, several other metrics have been highlighted, together with the accuracy obtained by random guessing (at 50%). The results are reported as accuracy (in %) (± standard deviation), where the accuracy is calculated using equation 5.1.

[2]Other values were explored. Higher values often led to the desired number of sentences being overridden to be able to still form three chunks, and lower values were found to lead to less meaningful topic representations.

Metric	Accuracy (%) (± standard deviation)
RMSE	46.42 (± 0.003)
Mean JS	47.01 (± 0.003)
Std JS	45.66 (± 0.004)
Oscillation	50.01 (± 0.002)
Random	50.00

Table 5.2: Accuracies on the sentence ordering task.

We observe that some metrics—such as Mean JS or RMSE—achieve accuracies in the 45%–47% range, which is worse than the 50% random baseline. The standard deviations (roughly 0.003–0.004) remain very small. The standard deviations are computed as the variation in accuracies across trials and it shows that, despite variations in individual permutations, the performance is consistent across these trials. One possible explanation for the accuracies dropping below 50% is that certain articles are thematically incoherent in their original form: it occasionally occurs that the model assigns high JS divergence scores to the original chunks. This can occur when the model struggles to extract a meaningful topic representation due to inconsistent writing present in the original article. Consequently, the permutations in such cases may fail to introduce a measurable spike in incoherence, as the baseline divergence already reflects significant thematic incoherence. This makes it challenging for the model to detect further disruptions caused by the permutations and, in quite some cases, lowers the measured incoherence.

The results align with the expectations mentioned in 4.1. Because our thematic coherence method focuses on fluctuations across larger portions of text, they are not sensitive to the sequential arrangement of sentences. Deviations occur, but the accuracies for each article are highly volatile and hence cannot discriminate consistently between coherent and permuted articles. The near-random performance of our measure on the sentence ordering task showcases its insensitivity to local sentence transitions.

5.1.5 Sentence replacement results

The results on the sentence replacement task will be discussed in this section. The RMSE is used as measure and the sentences per chunk is set at five. The accuracies across the different parameter configurations are shown in Table 5.3. Three trials were conducted per configuration setting, so the average and standard deviation of the accuracy across trials is reported. Each value is represented as accuracy (%) (± standard deviation).

% Chunks Affected	Number of Sentences Replaced				
	1 Sent.	**2 Sent.**	**3 Sent.**	**4 Sent.**	**All Sent.**
10%	56.87 (±1.36)	66.91 (±1.36)	70.06 (±1.43)	75.16 (±1.96)	78.21 (±0.23)
20%	61.22 (±0.17)	70.16 (±2.80)	74.04 (±0.45)	78.11 (±0.26)	80.61 (±1.16)
30%	65.02 (±2.30)	71.56 (±1.20)	77.52 (±1.10)	80.71 (±1.39)	82.36 (±1.01)
50%	68.72 (±0.57)	76.41 (±0.60)	79.76 (±0.44)	84.46 (±0.61)	87.05 (±1.10)
75%	70.62 (±0.99)	78.51 (±0.61)	84.46 (±1.82)	87.96 (±0.43)	87.97 (±0.57)
100%	72.61 (±1.70)	81.15 (±0.61)	87.30 (±1.30)	89.45 (±0.17)	90.10 (±1.23)

Table 5.3: Accuracies on the replacement task across different parameter configurations.

From these accuracies, a clear pattern emerges. Both as the percentage of chunks affected and the number of sentences of chunks replaced increase, the accuracies generally increase. This indicates a clear positive correlation between the level of disruption and the accuracy. To test whether the lowest observed accuracy of 56.9% is significantly better than random guessing, we perform a binomial test (under the null hypothesis that predictions are no better than random guessing [94]). At the $\alpha = 0.05$ significance level, the binomial test yielded a highly statistically significant result ($p < 0.001$), confirming the model's ability to detect even minor disruptions in thematic coherence. By extension, the statistical significance of the higher accuracies observed is also supported. The highest accuracy is obtained when all chunks are affected, and within those chunks, all sentences are replaced - in other words, when an article contains completely random sentences. The values from the table are visualized in Figure 5.1, with the standard deviations marked.

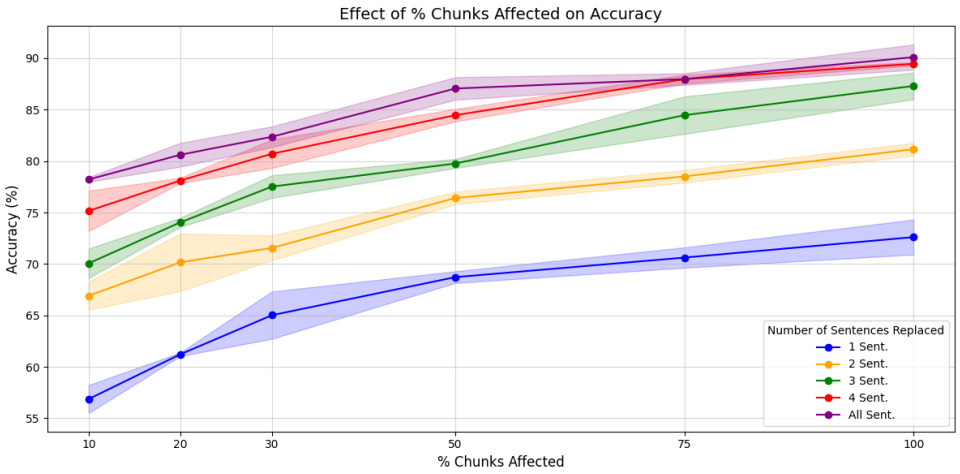


Figure 5.1: Accuracies across different parameter configurations (±standard deviations).

5.1.6 Discussion

Based on the analysis presented above, H1.1 is accepted. The observed metrics do not succeed in achieving accuracies higher than random guessing, demonstrating the model's inability to reliably distinguish between original and reordered texts. This is because the model focuses on aspects that go beyond the order of sentences and as a result, does not pick up on differences caused by reordering sentences.

H1.2 is accepted as the results have shown that the method can clearly discern thematically coherent from incoherent texts. Although the model accurately captures major disruptions, in real life, minor inconsistencies may be more common. The model demonstrates reasonable sensitivity to these but also shows that thematic coherence alone cannot fully capture overall text coherence. To fully capture this complexity, our model should be integrated with complementary measures that capture more fine-grained, syntactic or local coherence features. An example of this would be to create a parallel process to the thematic coherence computation that incorporates a local adjacency measure by, e.g. constructing an entity grid. We could then create a local coherence score based on for instance the density of this entity grid: a higher density of repeated and well-connected entities indicates a stronger local coherence [30]. We could then cleverly combine this score with our thematic coherence score, allowing our model to capture both thematic deviations and smaller sentence-level variations. Another observation that also stands out is that when all sentences in an article are replaced, the model achieves a 90.10% accuracy. This raises the question of why it is not detecting 100% of these fully replaced texts as less coherent. This is primarily due to the inherent interplay between the topic probability distribution of the full article and its chunks. When all sentences in all chunks are replaced with thematically dissimilar ones, the full article's topic probability distribution also changes accordingly. Since the full article consists of the chunks, each chunk contributes proportionally to the overall thematic structure—particularly in shorter texts with fewer chunks (e.g., in a three-chunk article, each chunk contributes approximately 33%). As a result, even fully randomly replaced chunks can create some degree of alignment with the new full article's topic distribution. This helps to explain why the replacement task does not achieve perfect accuracy under maximal disruption.

5.2 Real vs. fake news analysis

Having illustrated the proposed method can effectively capture thematic coherence, we will test its usefulness in distinguishing real from fake news articles. This experiment is related to research question RQ3 and intends to analyze two things: whether there is a significant difference in thematic coherence between real and fake news articles and to what extent we

can use the proposed method to accurately predict the veracity of articles. To answer this, the following experimental research questions are defined:

ERQ2.1 Is there a significant difference in thematic coherence between real and fake news articles?

ERQ2.2 Does the proposed thematic coherence method achieve significantly higher accuracy than random guessing in distinguishing real from fake news articles?

5.2.1 Dataset and pre-processing

The fake news dataset used in this research is the ISOT dataset [95]. We will consider some of its key characteristics first. The dataset is a compilation of several thousands truthful and fake human-written news articles. Real articles are obtained from different legitimate news sites (primarily Reuters), and fake news articles are retrieved from sites flagged as unreliable by Politifact.com. The articles report on different topics, with the majority of articles focused on political and world news topics. In order to make the dataset manageable, several pre-processing steps were taken. The fake articles, in particular, contained noise such as URLs, locations, editor's notes and source attributions. The fake dataset also contained 6,119 duplicates. The noise and duplicates were all removed such that only the actual article text remained. How these pre-processing steps are implemented can be found in the Github repository[3]. A more precise breakdown of the used dataset can be seen in Table 5.4.

News	Total articles	Type	Number of articles	Avg. sentences/article
Real	16,115	World news	7,598	19.34
		Politics news	8,517	
Fake	16,155	Government news	1,570	20.97
		Middle east	770	
		US news	775	
		Left news	4,457	
		Politics	6,838	
		News	9,050	

Table 5.4: Breakdown of the ISOT dataset.

5.2.2 Method

Having pre-processed the ISOT dataset, the next step is extracting topics from it (as outlined in phase 1 in Figure 3.1). To do so, BERTopic was fitted on the full dataset. This initial fitting yielded 353 topics. Such a high number of topics would lead to sparse comparisons of divergence values and make interpretation more challenging. Therefore, to ensure interpretability but to prevent documents from being forced into a cluster they do not belong in, the num-

[3]https://github.com/Willems-source/MScThesis

ber of topics was reduced to 50, using BERTopic's REDUCE function. The full list of topics is shown in the appendix (Table **??**). A much smaller number would risk oversimplifying the topic representations, whereas larger numbers would result in too fragmented topics. Also, excessive fragmentation means that many documents have near-zero probabilities on many topics, diluting the JS divergence values. After we have obtained the topic model containing the extracted topics, the steps in Algorithm 2 are performed, yielding a thematic coherence score. With this score, we will test two things in particular that are related to the experimental research questions.

5.2.2.1 Analyzing the thematic coherence of real and fake news articles

For the first part of the results and related to the first experimental research question ERQ2.1, we will analyze whether there is a significant difference between the thematic coherence of real and fake news articles. This is done by comparing the thematic coherence scores across all articles from the two groups. Whether this difference between the two groups is statistically significant will be assessed using a two-sample Welch's t-test [94]. We also quantify the effect size of this difference by measuring Cohen's d: $\frac{\text{mean difference}}{\text{pooled standard deviation}}$. Using the pooled standard deviation accounts for variability within both groups, providing a more robust effect size estimate. Cohen's d indicates the strength of the separation between the two groups, with values of $d = 0.2$, $d = 0.5$, and $d = 0.8$ typically representing small, medium, and large effect sizes, respectively. Higher values of d in our research thus signify a stronger distinction between the thematic coherence of real and fake news articles.

5.2.2.2 Classifying real vs. fake news articles

After we have investigated the differences in thematic coherence between real and fake news articles, we can move on to assessing the use of the method in predicting whether an article is real or fake. This pertains to answering experimental research question ERQ2.2. To do so, two simple classifiers outlined below will be tested:

- **Threshold classification.** The first classification method we explore uses a threshold on the coherence score. A threshold provides a cutoff for classifying articles as real or fake. To test the classifier, we first randomly partition the data into a training (70%) and testing (30%) set. We use the training set to train the classifier by tuning its threshold parameter. Different thresholds are explored to find the one that yields the highest accuracy. More specifically, thresholds ranging from 0 to 1 were explored in increments of 0.005 to compromise between detail and runtime. Once the optimal threshold for the training set is determined, we apply it to the test set to obtain the final accuracy. An article is classified as real if it has a coherence score that is lower than the threshold, indicating lower

thematic deviation, and classified as fake if the coherence score exceeds this threshold. To assess the reliability of the results, we perform a binomial test to determine whether the test set accuracy is significantly higher than the 50% baseline expected from random guessing. Finally, Cohen's d will also be calculated to measure the effect size.

- **Logistic regression.** Threshold classification provides a simple baseline, but it assumes a hard binary separation in coherence scores that may not reflect the actual distribution of the data. The second method that will therefore be tested is logistic regression. Logistic regression differs from threshold classification by modeling the probability of an article being real or fake as a function of the coherence scores. Instead of forcing a strict cutoff, it considers the overlap between real and fake articles and gives probabilities. This helps it handle cases near the threshold better, where coherence scores are too close to clearly separate real from fake. For the logistic regression, we also split the dataset into a training (70%) and test (30%) set. The model uses the RMSE value for the coherence score as its sole feature. We evaluate the accuracy on the test and provide the classification report and confusion matrix. Additionally, we analyze the importance of RMSE as a feature. The statistical significance of the RMSE coefficient is assessed using a Wald test [94], and the overall accuracy is tested against random guessing through a binomial test. Both tests are performed to ensure that the model's performance is significantly different from random chance. Finally, we also quantify the effect size using Cohen's d.

These methods intend to answer the research questions posed at the beginning of this section. The corresponding hypotheses, also motivated by the findings from section 2.3, are as follows:

H2.1 Real news articles exhibit significantly higher thematic coherence compared to fake news articles.

H2.2 The proposed thematic coherence method achieves significantly higher accuracy than random guessing in distinguishing real from fake news articles.

5.2.3 Results

The results are similarly divided into two subsections. The first subsection will discuss whether there is a significant difference in the thematic coherence of real and fake news articles, while the second subsection will evaluate the classification performance in predicting whether an article is real or fake.

5.2.3.1 The thematic coherence of real and fake news articles

The average coherence scores (calculated by applying Algorithm 2) for the real and fake news articles are displayed in Table 5.5. Note that a higher score means a higher thematic divergence and thus, lower thematic coherence. The table provides an overview of the average scores for the full dataset, as well as the scores for real and fake articles. Values are reported as average ± standard deviation.

Thematic Coherence Score	Full dataset	Real Articles	Fake Articles
Value	0.398 ± 0.208	0.363 ± 0.209	0.434 ± 0.201

Table 5.5: Overview of the average coherence scores for the ISOT dataset.

From Table 5.5, we find that fake news articles tend to have a higher average score (0.434) compared to real news articles (0.363), indicating that fake articles exhibit lower thematic coherence on average. The standard deviations (0.201 for fake and 0.209 for real articles) suggest considerable variability within each class, leading to some overlap in scores between real and fake articles. Conducting Welch's t-test reveals that the difference is highly statistically significant ($p < 0.001$). To quantify the strength of the separation between the two groups, we calculate Cohen's $d = -0.346$. The negative sign indicates that real articles have lower coherence scores (meaning higher thematic coherence) than fake articles. The magnitude of d indicates a small to medium effect size, as $0.2 < |d| = 0.346 < 0.5$.

5.2.3.2 Classification performance of real vs. fake news articles

The observed thematic coherence scores for real and fake news articles suggest there is a significant difference between the two groups. Building on this, we evaluate the utility of thematic coherence in classification, using the coherence scores to distinguish real from fake articles. The results of the two tested classifiers are presented below.

Threshold classification. For the threshold classification, the optimal threshold score of 0.410 was determined on the training set. When applied to the test set, this threshold achieved an accuracy of 56.41%. The confusion matrix displaying the actual versus the predicted labels can be found in Figure 5.2. Using the binomial test on the test set accuracy, a p-value of $p < 0.001$ was obtained, indicating the model performs significantly better than random guessing in classifying real and fake documents at the $\alpha = 0.05$ level. In quantifying the effect size, we find that Cohen's $d = 0.1276$, indicating a small effect size.

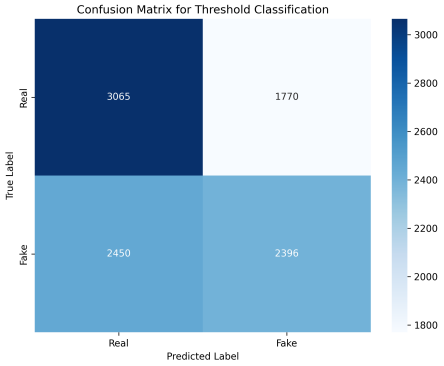


Figure 5.2: Confusion matrix for threshold classification.

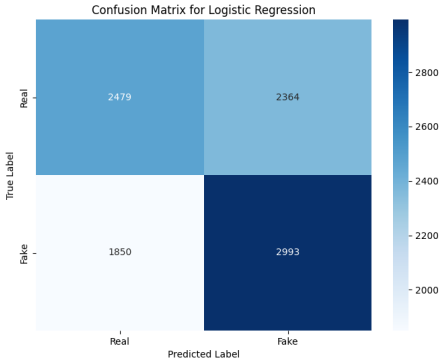


Figure 5.3: Confusion matrix for logistic regression.

Logistic regression. Figure 5.3 shows the confusion matrix for the logistic regression model, visualizing the actual versus predicted labels. Using RMSE as the only feature, this logistic regression model provides a baseline for performance. The classification report can be seen in Table 5.6 below. The RMSE coefficient (1.6493) was found to be statistically significant ($p < 0.001$) using a Wald test, confirming its importance as a predictor. Specifically, a 1-unit increase in RMSE raises the odds of an article being classified as fake by approximately 5.2 times ($e^{1.6493}$).

Class	Precision	Recall	F1-Score	Support
Real (0)	0.57	0.51	0.54	4834
Fake (1)	0.56	0.62	0.59	4843
Accuracy		0.57		9677
Macro Avg	0.57	0.57	0.56	9677
Weighted Avg	0.57	0.57	0.56	9677

Table 5.6: Classification report for real vs. fake news.

We performed the binomial test to quantify whether the observed accuracies are significantly different from random guessing. The observed accuracy of the classifier was 56.62% with $p < 0.001$. This indicates that the classifier's performance is statistically significantly better than random guessing at the $\alpha = 0.05$ level. Finally, Cohen's d gives a value of 0.1304, indicating a small effect size compared to random guessing. While the improvement is statistically significant, the overall accuracy of 56.62% is still modest in absolute terms.

5.2.4 Exploring impact of method components

The results so far have highlighted meaningful differences in the thematic coherence of real and fake news articles. Specifically, real news articles were found to exhibit significantly higher thematic coherence than fake articles, reinforcing the potential of thematic coherence as a valuable distinguishing feature.

It is important to consider that different components in the proposed method may influence the results. For instance, in the previous classification experiment, topics were extracted from the ISOT dataset (relating to phase one, as discussed in section 3.1) and the thematic coherence computation (relating to phase two, discussed in section 3.2) was applied to articles from this same dataset. To enhance the understanding of the proposed method, it is also interesting to investigate what happens if we apply these topics to a fake news dataset with articles from a different domain. The rationale is to assess whether domain-specific topics extracted from one dataset can still capture relevant thematic signals when transferred to a new domain, where misleading information may be conveyed in a different writing style. Likewise, we investigate how topics extracted from a more generic dataset behave on these fake news tasks. This may reduce cross-interference - merging similar topics on e.g. politics into one, not forcing the model to choose between multiple closely related topics - and ensure topics are more transferable across datasets. However, it may also miss the granularity needed to capture subtle, domain-specific patterns.

This section intends to explore these different scenarios and give a concise overview of the results under these different configurations. First, the datasets used for exploration will be discussed. The method is identical to the previous experiment, so the overview of the results for the different scenarios will be presented immediately after. This overview contains the results for the previous experiment, to which the results for three new experiments will be added:

- The ISOT topics applied an out-of-domain fake news dataset.
- Generic topics applied to the ISOT dataset.
- Generic topics applied to an out-of-domain fake news dataset.

5.2.4.1 Datasets and pre-processing

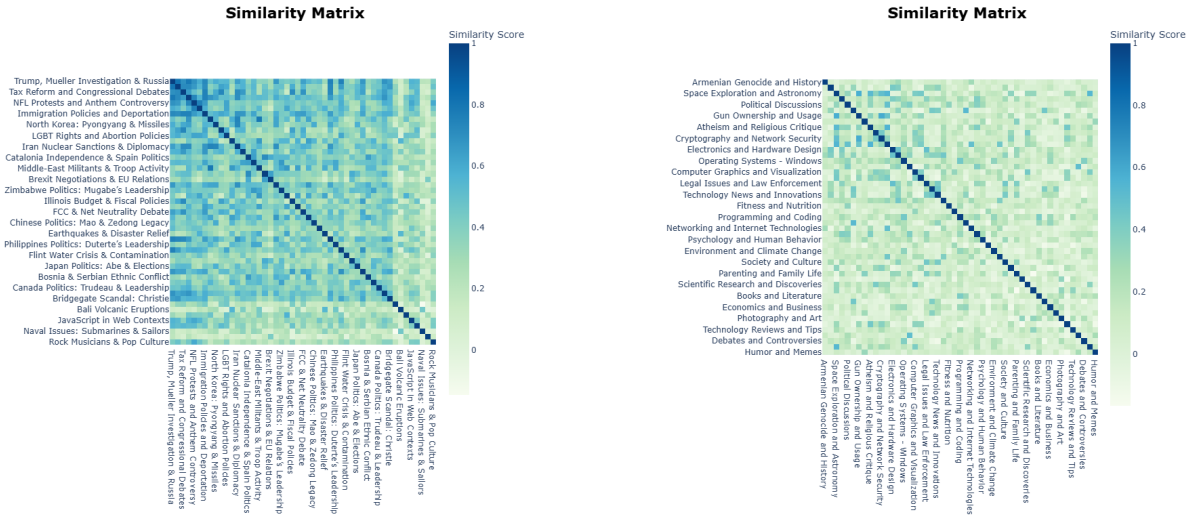
A domain that is also often targeted by fake news sources is health and well-being, leading to the spread of false information on medical treatments or public health measures, as mentioned in section 1.1. To test the transferability of the topics extracted from the ISOT dataset and their ability to perform on articles discussing different themes, the procedure outlined earlier in this section will be applied to the FakeHealth dataset. The overview statistics for the FakeHealth dataset are shown in Table 5.7.

Label	Total articles	Type	Number of articles	Avg. sentences/article
Real	1,464	HealthRelease	307	31.17
		HealthStory	1,157	31.49
Fake	742	HealthRelease	285	28.06
		HealthStory	457	22.23

Table 5.7: Breakdown of the FakeHealth dataset, including average sentences per article.

The dataset used consists of two separate datasets, HealthStory and HealthRelease, corresponding to news stories (reported by news media such as Reuters Health) and releases (from various institutes including universities and research centers) [96]. Given their similar form and length, they can be used directly for this robustness analysis. The articles are assessed based on several criteria that measure the degree of overclaiming, missing information, and the reliability of sources. Scores are given by at least two experts and the overall score ranges from 0 to 5, based on the number of criteria satisfied. Following previous work [97] [96], an article is assigned the label fake if the overall score is lower than 3 and real otherwise.

On the topic extraction side, a different option is also explored by extracting topics from a generic dataset. A well-suited dataset for this is the 20 Newsgroups collection. This dataset is a widely used benchmark in topic modeling due to its diversity in topics discussed [98]. Extracting 50 topics from this ensures consistency with our previous analyses, striking the same balance between interpretability and meaningful topics without them being overly dense. BERTopic initially extracted 130 topics from the dataset, so the number of topics still had to be reduced significantly using BERTopic's REDUCE function like before. The full list of topics is included in the appendix (in Table 3). From Figure 5.4b, it follows that the topics are more dissimilar (as indicated by the lighter colors) than the topics extracted from the ISOT dataset (in Figure 5.4a), which shows its suitability as a generic benchmark.



(a) Similarity matrix for the ISOT topics.

(b) Similarity matrix for the generic topics.

Figure 5.4: Similarity matrices for the ISOT topics (left) and the generic topics (right).

5.2.4.2 Results

First, we analyze the thematic coherence of real and fake news articles in this new health-domain dataset. The coherence scores for the FakeHealth dataset are displayed in Table 5.8. Values are again reported as average ± standard deviation.

Thematic Coherence Score	Full dataset	Real Articles	Fake Articles
Value	0.481 ± 0.176	0.484 ± 0.171	0.475 ± 0.186

Table 5.8: Overview of the average coherence scores for the FakeHealth dataset.

From Table 5.8, it becomes apparent that real and fake news articles in the FakeHealth dataset exhibit very similar thematic coherence scores. We see that real articles show a slightly higher average score (0.484) compared to fake articles (0.475), indicating marginally lower thematic coherence for real articles. The standard deviations (0.171 for real and 0.186 for fake articles) suggest comparable variability within both groups. Conducting Welch's t-test confirms that the difference in scores is not statistically significant ($p = 0.275$). To quantify the strength of the separation between the two groups, Cohen's $d = 0.051$ was calculated. The small magnitude of d indicates a negligible effect size, further suggesting that thematic coherence scores struggle to distinguish real and fake news articles in this health-domain dataset. This observation aligns with the idea that the labeling criteria for the FakeHealth, as discussed in section 5.2.4.1, which are based on factual correctness rather than linguistic differences, may limit the ability of our method to differentiate between real and fake news articles.

Second, we compare the performance in the four scenarios on the classification tasks. Table 5.9 shows the results for the different experiment configurations across the two classification

tasks. Accuracies for threshold classification are reported on the test set and the p-values are accompanied by an asterisk (*) if they are significant at the $\alpha = 0.05$ level.

Topics Extracted From	**Applied To Dataset**	**Threshold**			**Logistic Regression**		
		Acc. (%)	p-val	Cohen's d	Acc. (%)	p-val	Cohen's d
ISOT	ISOT	56.4	$< 0.001^*$	0.13	56.6	$< 0.001^*$	0.13
	FakeHealth	51.2	0.283	0.02	51.6	0.173	0.03
Generic (20 NG)	ISOT	52.0	$< 0.001^*$	0.04	51.7	$< 0.001^*$	0.03
	FakeHealth	49.4	0.670	-0.01	51.8	0.156	0.04

Table 5.9: Results for different topics applied to different datasets.

The table demonstrates that topics extracted from the ISOT dataset yield statistically significant results for ISOT using both threshold classification and logistic regression, achieving 56.4% ($p < 0.001^*$) and 56.6% ($p < 0.001^*$), respectively. However, for the FakeHealth dataset, both threshold classification (51.2%, $p = 0.283$) and logistic regression (51.6%, $p = 0.173$) do not achieve statistical significance. When generic topics are applied, the performance is weaker across both datasets. For ISOT, threshold classification achieves 52.0% accuracy ($p < 0.001^*$) and logistic regression achieves 51.7% accuracy ($p < 0.001^*$). For FakeHealth, neither threshold classification (49.4%, $p = 0.670$) nor logistic regression (51.8%, $p = 0.156$) achieves statistically significant results, highlighting the limitations of using generic topics in this task.

5.2.5 Discussion

Overall, several experiments have been conducted related to investigating the thematic coherence of real and fake news articles. We will first reflect on the average differences between real and fake news in terms of thematic coherence. Then we will discuss the results on the classification tasks. Following this, we will reflect on the method's performance on an out-of-domain datasets and using generic topics.

First, we can accept H2.1 for the ISOT dataset: there is a statistically significant difference in thematic coherence between real and fake articles. On average, real news exhibits lower scores (i.e., higher thematic coherence) than fake news (0.363 for real compared to 0.434 for fake). This result alone is noteworthy: it suggests that, at least for news articles related to politics and world events, fake news may show greater topical divergence within a single article. A small-to-medium effect size (as indicated by Cohen's $d = 0.346$) further indicates that a clear separation between these two groups exists, and the difference was found to be significant.

Building on these observed differences, we tested simple threshold and logistic regression

classifiers that used the thematic coherence score as the sole predictor. The results on these classifiers lead us to also accept H2.2 for the ISOT dataset. Both approaches yielded results statistically better than random guessing. The threshold classifier and logistic regression obtained an accuracy of 56.41% and 56.62%, respectively. This shows that both provide some discriminatory power - meaning they can identify some patterns in thematic coherence differences - to distinguish real from fake news articles. However, merely computing the thematic coherence of a text is obviously not sufficient for determining the veracity of a text on is own. This is also supported by the small effect sizes for both classifiers (0.1276 and 0.1304 for threshold classification and logistic regression respectively). Yet for the ISOT dataset, this is evidence that thematic coherence can be a valuable additional indicator for identifying fake news. Especially local coherence features, such as the number of entity transitions or lexical overlap between adjacent sentences, which operate on a sentence-to-sentence level, could well be complemented by our thematic coherence measure for improved performance.

Second, we tested the same (ISOT-extracted) topic model to the FakeHealth dataset. For the FakeHealth dataset, coherence scoring was found to be unuseful in distinguishing real from fake news articles. Both the difference in average scores between the real and fake news articles, as well as the performance of the classifiers, were insignificant. This leads to near-random differences between the groups and near-random performance on classification, making us reject both H2.1 and H2.2 for the FakeHealth dataset. Such findings support the observation that while the approach shows potential in contexts where content deviates thematically (e.g., broader socio-political topics like in the ISOT dataset), it may not generalize well to domains highly centered around a few related topics. In addition, the nature of the FakeHealth dataset also contributes to these results: real and fake labels are assigned based on the degree of overclaiming or the reliability of sources in articles that are otherwise well-written. Such cases limit the use of our coherence-based method.

Third, we extracted topics from a generic dataset to detect fake news in both the ISOT and the FakeHealth datasets. For ISOT, the results still showed similar patterns to before, but far less pronounced (classification accuracies of ~52% and effect sizes of $d = 0.03 - 0.04$). For FakeHealth, the generic model did not yield any significant results. This outcome suggests that generic topics lack the granularity needed to capture subtle inconsistencies. Where topics extracted from the ISOT dataset may consider multiple different topics on politics, a generic topic model may only discern one, decreasing its ability to detect thematic incoherence. Consequently, the coherence-based detection method appears more effective in broader contexts, as this gives more opportunity to vary in the topics of a text.

5.3 Human-written vs. LLM-generated news analysis

The experiments above have solely focused on human-written articles. Nowadays, when investigating the style of news articles, it is crucial to incorporate text generated by LLMs in the analysis. To that end, the thematic coherence of LLM-generated news articles will be investigated in this section. This final experiment follows a similar structure as the fake news detection experiment: first investigating the difference between human-written and LLM-generated news articles, followed by assessing the ability of the method to classify articles into one of the two groups.

Previous studies have used techniques such as conditioning generation on knowledge elements or adversarial reinforcement learning [99] to generate synthetic fake news. These, however, require expertise and costly designs to generate text. Others have used LLMs to generate fake news using e.g. structured mimicry prompting, a structured way to generate similar texts. This may however significantly alter the distribution similarity between the news articles [61] and risks blurring the line between real and fake across LLM-generated and human-written articles. Consequently, we omit the veracity dimension for this experiment to focus solely on the effects of LLM generation.

Based on the findings outlined in section 2.5, we hypothesize that LLM-generated text exhibits a lower degree of thematic coherence. This experiment is related to research question RQ4 and leads to the following experimental research questions and hypotheses:

ERQ3.1 Is there a significant difference in thematic coherence between human-written and LLM-generated news articles?

ERQ3.2 Does the proposed thematic coherence method achieve significantly higher accuracy than random guessing in distinguishing human-written from LLM-generated news articles?

H3.1 Human-written news articles exhibit significantly higher thematic coherence compared to LLM-generated news articles.

H3.2 The proposed thematic coherence method achieves significantly higher accuracy than random guessing in distinguishing human-written from LLM-generated news articles.

5.3.1 Dataset

The LLM-generated text comes from the MAGE dataset [100]. It is an extensive dataset containing text from different domains such as opinion statements, question answering, story generation and news articles. This research will focus on the news articles. The authors created the LLM-generated text by using three different prompting strategies:

- Continuation prompts: they prompt the LLM to finish a human-written text after providing the first 30 words;
- Topical prompts: they prompt the LLM to write a story on a given topic;
- Specified prompts: they prompt the LLM to create a text with information from a specified source.

In the original dataset, 27 LLMs are included. For this research, including all was not considered necessary. In the fake news domain, malicious actors will turn to models that are a compelling option both economically and in terms of accessibility. Therefore, OpenAI's GPT models are chosen first to simulate a more real-life journalistic scenario [61]. To still cover some of the diversity in text styles produced by other architectures, the texts generated by Meta's open-source model LLaMA are included too. The relevant summary statistics can be seen in Figure 5.10. For the human-written articles, the real news articles from the ISOT dataset were used (shown in 5.4). To reiterate: this means we have merged the articles (from both models) from Table 5.10 and the real articles from Table 5.4 for this experiment into one dataset, which will be referred to as the LLM dataset from now on.

Model	Variants	Prompt Strategy	Articles	Avg. Sentences/article
GPT (OpenAI)	text-davinci-002	Continuation	748	14.99
		Specified	697	8.83
		Topical	618	8.55
	text-davinci-003	Continuation	614	8.62
		Specified	999	11.61
		Topical	999	10.79
	gpt-turbo-3.5	Continuation	174	7.47
		Specified	1000	14.59
		Topical	999	12.68
LLaMA (Meta)	6B	Continuation	897	23.46
	13B	Continuation	879	21.96
	65B	Continuation	885	20.96

Table 5.10: Summary statistics of the used part of the MAGE dataset.

5.3.2 Method

Following the procedure outlined in the previous experiments, we first analyze the coherence scores of the full dataset, investigating possible differences in thematic coherence between human-written and LLM-generated articles. To come to these scores, we use the generic topic model described in 5.2.4 to run Algorithm 2. This topic model is used to ensure a fair evaluation by avoiding biases toward the specific topics discussed in either human-written or LLM-

generated articles. By relying on a generic topic model trained on a diverse dataset, the analysis minimizes the influence of any particular topic distribution that might skew the results. For the classification, again a threshold classifier and a logistic regression are tested. Since it is hypothesized that human-written articles will exhibit higher thematic coherence, articles are classified as 'human-written' if their coherence score is below or equal to the threshold, and as 'LLM-generated' otherwise.

5.3.3 Results

First, we analyze the thematic coherence of human-written and LLM-generated news articles. The coherence scores for the LLM dataset are displayed in Table 5.11 below.

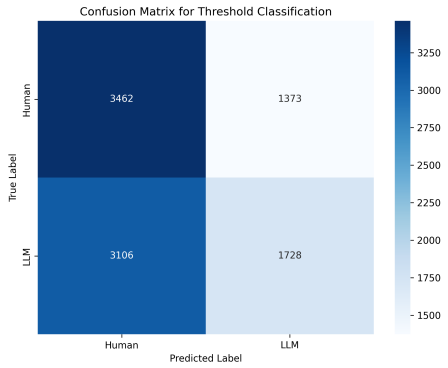
Thematic Coherence score	**Full dataset**	**Human-written Articles**	**LLM-written Articles**
Value	0.443 ± 0.192	0.439 ± 0.179	0.450 ± 0.211

Table 5.11: Overview of the average coherence scores for the LLM dataset.

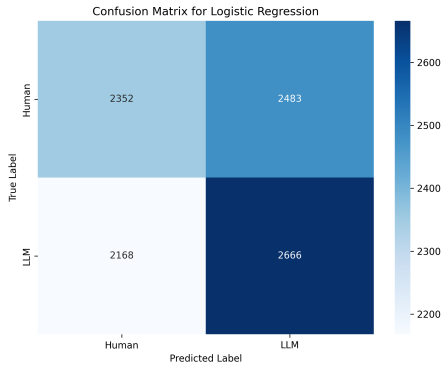
Table 5.11 shows us that LLM-generated news articles tend to have slightly higher scores (0.450) compared to human-written articles (0.439), indicating that LLM articles exhibit marginally lower thematic coherence on average. The standard deviations (0.211 for LLM and 0.179 for human articles) indicate overlapping variability in scores between the two groups. Conducting Welch's t-test reveals that the difference is highly statistically significant ($p < 0.001$). Cohen's $d = 0.061$ tells us that the effect size is very small.

Second, the results for the classifiers are displayed in Figure 5.5. For threshold classification, the best threshold on the training set was found at a coherence score of 0.55. Using this threshold, we obtain an accuracy of 53.68% on the test set. Using the binomial test, ($p < 0.001$) was obtained, demonstrating that the classifier performs significantly better at distinguishing LLM-generated from human-written documents than random guessing at the $\alpha = 0.05$ significance level. Cohen's d = 0.0735, meaning a very small effect size.

The relationship between actual and predicted labels for the logistic regression is shown in Figure 5.5b. The evaluation metrics for this model are summarized in Table 5.12. The RMSE coefficient (0.2533) in the model was found to be statistically significant ($p < 0.001$) using a Wald test. Specifically, a 1-unit increase in RMSE increases the odds of labeling a text as LLM-generated by approximately 1.29 times ($e^{0.2533}$). The logistic regression achieves an overall accuracy of 51.90%, which is statistically significantly better than random chance ($p < 0.001$). Cohen's d = 0.0380 indicates a very small effect size. The precision, recall, and F1 scores are evenly distributed across both classes, reflecting the model's balanced but limited ability to differentiate between human and LLM-generated text.



(a) Confusion matrix for threshold classification.



(b) Confusion matrix for logistic regression.

Figure 5.5: (a) Confusion matrix for threshold classification and (b) Confusion matrix for logistic regression.

Class	Precision	Recall	F1-Score	Support
Human-written (0)	0.52	0.49	0.50	4835
LLM-generated (1)	0.52	0.55	0.53	4834
Accuracy		0.52		9669
Macro Avg	0.52	0.52	0.52	9669
Weighted Avg	0.52	0.52	0.52	9669

Table 5.12: Classification report for human-written vs. LLM-generated news.

Given that two different LLMs and three different prompting strategies are used, comparing them can potentially tell us something about the behavior of LLMs. Particularly interesting is the effect of the prompting strategy on the obtained coherence scores, as this can help us better understand the behavior of LLMs under different instructions. This is done only for the GPT-generated texts, as the LLaMA-generated texts are only generated using the continuation strategy and may therefore skew the results. When analyzing the coherence scores for different prompting strategies, the continuation prompting strategy yields the highest value (0.454), followed by specified (0.439) and topical (0.431). The differences between continuation on the one hand and specified and topical on the other hand (p=0.093 and p=0.003 respectively) are statistically significant at the 10% significance level. This is a first indication that adding constraints by telling the model to either write using a specified source or to write on a given topic increases thematic coherence compared to instructing it to freely continue a certain snippet.

5.3.4 Discussion

Based on the first experiment that analyzes the thematic coherence of human-written and LLM-generated news articles, H3.1 is accepted. The difference in thematic coherence between human-written and LLM-generated news articles was found to be significant: the scores indicated LLM-generated text is slightly more thematically incoherent on average. Similarly, H3.2

is also accepted. However, in both the broad analysis and the classification tasks, the effect sizes are very small. This is reflected by the values for Cohen's d not exceeding 0.08 (while 0.2 is still considered a small effect size) and the both classifiers obtaining a modest accuracy of 52%. These findings suggest that while thematic coherence provides some discriminatory power, it alone does not suffice for robust detection of LLM-generated text.

One possible explanation is the token-level training objective (also mentioned in section 2.5). This causes LLMs to maintain surface-level continuity quite well by not abruptly jumping from one domain to the other. This would also mean that the topics extracted for each chunk remain fairly stable. Another potential reason is the fact that the LLM-generated texts are affected by their prompts, potentially limiting the opportunities for big theme shifts. We have seen a first indication that if an LLM is prompted to use a specified source or write on a given topic, it thematically adheres to that more closely than when it can freely finish a given snippet. Conversely, a long human-written article may explore multiple angles causing it to show gradual thematic drift. This may be less likely to happen if a clearly defined prompt is given to an LLM.

These results underscore the importance of carefully considering the limitations of thematic coherence as a singular feature for distinguishing human-written and LLM-generated texts. However, the statistically significant findings still highlight its value as part of a broader detection framework.

5.3.5 Comparison across all classes

Having conducted all of our experiments, we can put everything together for a final comparison of all thematic coherence metrics (as defined in section 3.2.5). To that end, the average values for human-written real news, human-written fake news, and LLM-generated text are shown in Figure 5.6.

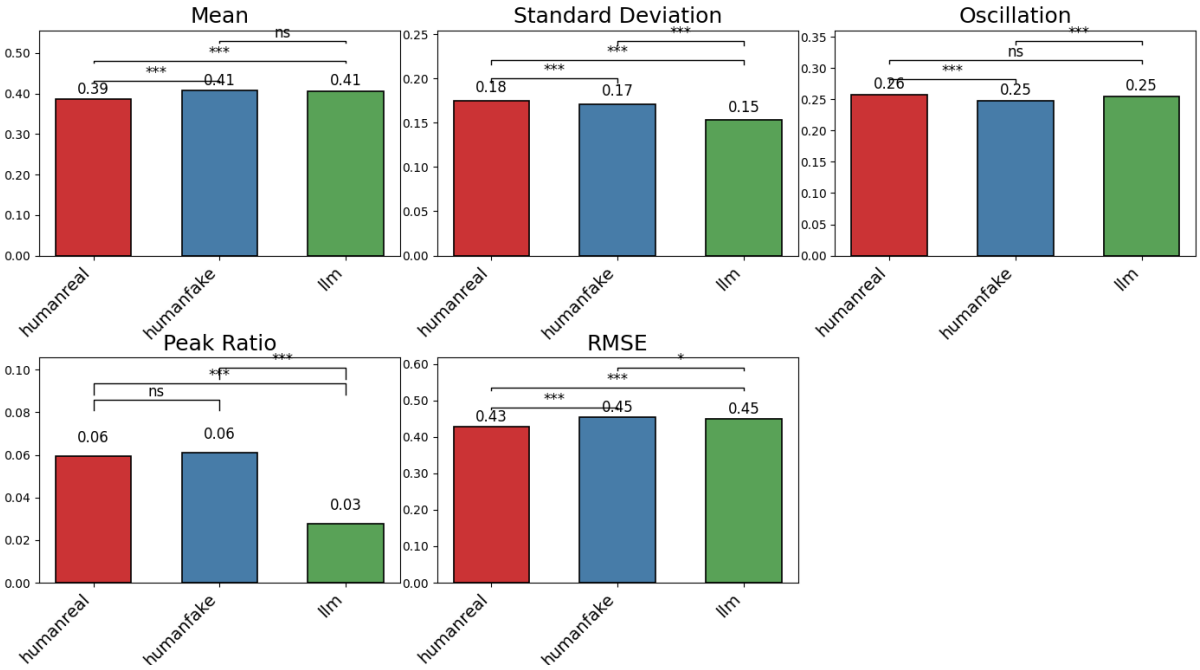


Figure 5.6: Average metrics across the three investigated classes.

The metrics from left to right are the mean, standard deviation, oscillation, peak ratio and the RMSE. The different classes are represented by different colors: human-written real news (red), human-written fake news (blue), and LLM-generated news (green). Whether the differences are statistically significant or not, as measured by a pairwise t-test, is indicated by the arc connecting the two groups. If p<0.01, the difference is assigned ***, followed by ** for p<0.05 and * for p<0.1. The label 'ns' indicates no significant difference. We have obtained the values using the generic topic model to ensure an unbiased comparison. Note that higher values for the metrics mean lower thematic coherence. Based on this figure, we make two key observations:

- The RMSE indicates thematic coherence is significantly higher in human-written real news articles compared to their fake counterparts or to LLM-generated text ($p < 0.001$). This finding suggests that, on average, in a real article a tighter thematic structure is maintained. The effect sizes for the RMSE for human-written real versus human-written fake and LLM-generated are 0.09 and 0.06. Similarly, for the mean, they are 0.11 and 0.10, respectively. All four values indicate small effect sizes.
- LLM-generated text shows noticeably less thematic variability, indicated by the lower standard deviation and the lower peak ratio. This is to some extent (especially for the lower peak ratio) attributable to the shorter article length of the GPT-generated articles, but this observed effect is still highly significant (p<0.001) for only considering the LLaMA-generated articles, which in fact have a higher average amount of sentences per

articles than both human-written classes. This implies that LLMs produce text that is consistent but slightly less thematically coherent overall. The effect size for the standard deviation of LLM-generated text compared to human-written real and fake articles is 0.20 and 0.17 respectively, indicating a small effect size. For the peak ratios, they are 0.32 and 0.34 respectively, indicating a small-to-medium effect size.

In short, human-written real articles generally rank as the most thematically coherent, as indicated by the lowest mean and RMSE. LLMs have shown to exhibit the smoothest chunk-level transitions - demonstrated by the low standard deviation and peak ratio. Human-written fake news closely resembles LLM-generated text in terms of overall thematic divergence, as indicated by the minimal or non-significant differences in mean and RMSE scores between the two classes.

5.4 Explanation evaluation

Finally, we will examine the generated explanation to uncover potential supplementary insights. These insights aim to deepen our understanding of the method and shed light on opportunities for future refinements, which we will further expand upon in section 6.3. The generated explanation is visual in nature and accompanied by some short explanatory texts to best highlight the model behavior. The explanation is focused on the top three most important topics in the full article and the chunks rather than being an exhaustive analysis. The explanation helps us to better understand the thematic progression throughout an article. Moreover, it needs to be noted that this experiment uses the topics extracted and the articles from the ISOT dataset. An example explanation for a randomly sampled article from the ISOT dataset is given below in Figure 5.7. To explore how the method generates explanations across different inputs, we use an article distinct from the running example in section 3.2.

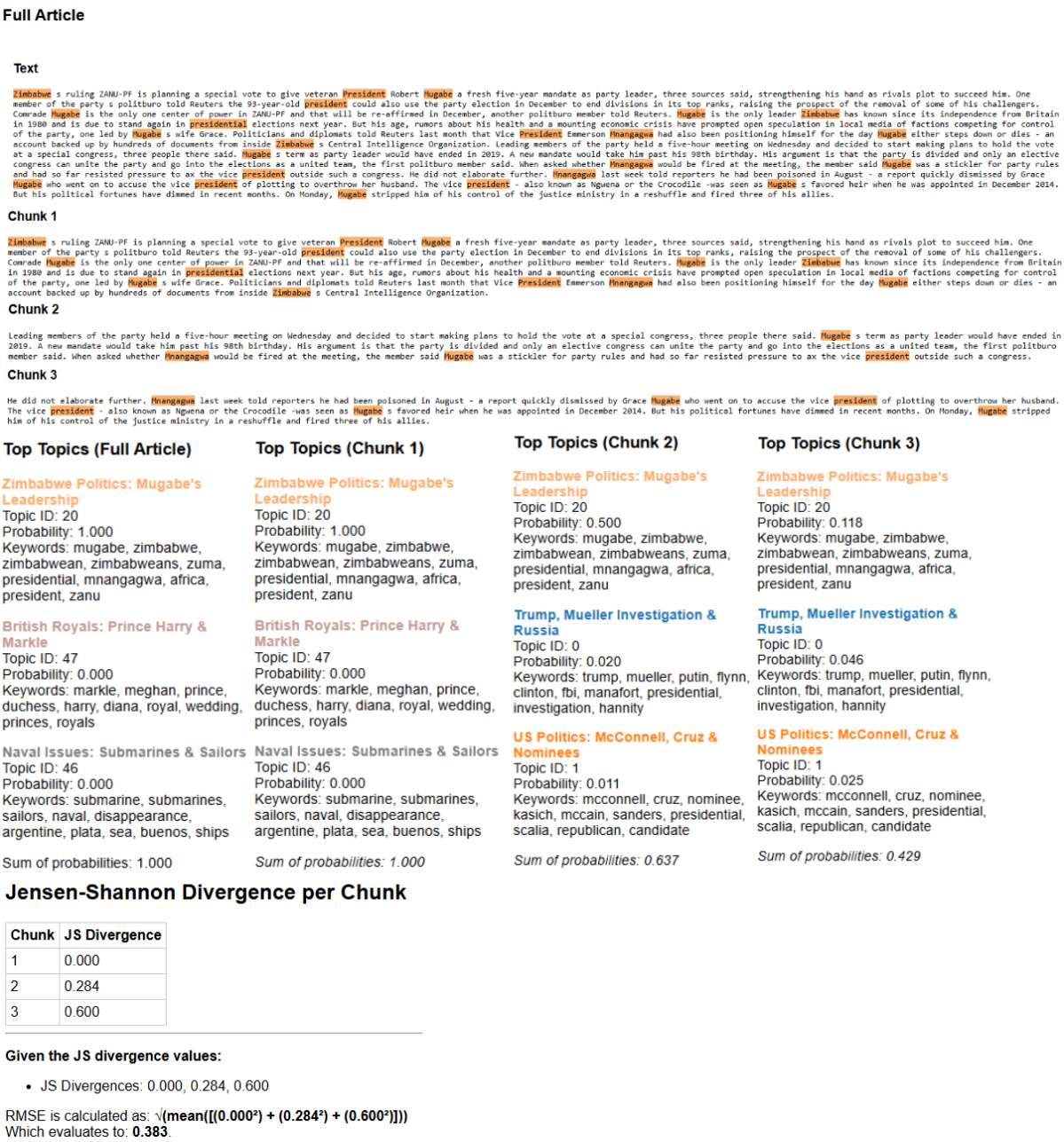


Figure 5.7: Example of a generated explanation to demonstrate model behavior.

The explanation is generated by running Algorithm 3. For this analysis, it is good to restate that if the sum is not equal to one, it indicates the model is unsure about a portion of the topics in the text, and the remainder is assigned to the outlier topic. In the provided example, we see that there are many keywords related to the Zimbabwe Politics topic highlighted in the text, already signaling to the user that this is a central topic. The other steps are the same as explained in section 3.2.6 and, in this example, result in a coherence score of 0.383, indicating reasonable thematic coherence.

Based on the example outlined above, three interesting observations that explain certain workings of the method can be made:

- There is a correlation between the confidence of the model in assigning topics to a text and the length of that text. Even though each chunk contains 5 sentences (with the exception of chunk 1 containing 6), our model does not guarantee the sentences are equal in length, which is the case in the shown explanation. We see that the sum of probabilities decreases from chunk 1 to 3. By experimenting with sentence lengths within a chunk, we found that this is partially due to the amount of information present in the respective chunks decreasing too.
- In chunks 2 and 3, a key characteristic of the model's behavior emerges. From the assigned probabilities, it is still clear that the model considers Zimbabwe Politics the most probable topic by some distance. However, it is no longer with a 100% certainty, like it was for the full article and chunk 1. What we see happening instead is that after the Zimbabwe Politics topic, the model assigns the two most common topics in the dataset as the next most probable topics. Given that the model is designed to maximize the likelihood, choosing the most common topics is a safe bet that aligns with probabilities learned during the extraction process [79]. This may however impact the overall coherence value in one of two ways. If the rest of the article is relatively aligned, which is the case in the highlighted example, it may artificially inflate the divergence value. On the other hand, if both the full article topics and the chunk topics contain a higher degree of uncertainty, the assignment of the most common topics may keep the divergence values artificially low as alignment between the full article and chunks is detected, even though that may not really reflect the semantic content.
- Finally, overlapping keywords may occur because BERTopic extracts representative words for each topic based on their importance. This may lead to shared keywords. Given that the ISOT dataset is rather centered around politics, one such keyword that's present in multiple topics is 'president'. The presence of overlapping keywords may impact results by reducing the model's ability to distinguish between related topics. This further illustrates the importance of balanced training data for topic extraction to mitigate this bias.

6. Discussion and conclusion

The previous chapter presented the findings with a brief discussion of each conducted experiment. In this chapter, the overall results will be discussed, put in broader perspective and the research questions will be answered. Following that, some limitations of this research and future research direction will be shared.

6.1 Overview of the results

The proposed method has been evaluated on two coherence evaluation tasks. We have shown that for thematic coherence - which is one aspect of coherence - the existing sentence ordering task is not informative, as illustrated by the accuracies around ~50% on the respective task. This confirmed the hypothesis that the sentence order minimally affects thematic coherence, showcasing the need for a second evaluation task. The performance on the newly proposed sentence replacement task demonstrates that the proposed method can capture thematic disruptions effectively and shows that it should be considered as an additional evaluation task in future research on coherence modeling.

The experiments that followed the coherence evaluation tasks have revealed clear patterns in thematic coherence across real news, fake news, and LLM-generated news articles. Real news consistently demonstrated higher thematic coherence than fake news, as evidenced by significantly lower coherence scores (0.363 for real vs. 0.434 for fake). This statistically significant difference, reflected by a small-to-medium effect size (Cohen's $d = 0.346$), is a key insight of our study. It suggests that, at least for news articles discussing broader domains, fake news articles show greater topical divergence within a single article. Classification using thematic coherence provided modest discriminatory power, with both threshold classification and logistic regression achieving accuracies of approximately 56%. These results support the use of thematic coherence as a supplementary indicator for identifying fake news. Its utility is likely especially strong when combined with local coherence features (such as entity grids or semantic similarity between adjacent sentences), as this integration can provide a more comprehensive assessment of coherence. When applied to the FakeHealth dataset, coherence scores could not distinguish between real and fake articles, with scores for both groups being nearly identical (0.484 vs. 0.475). This underscores the method's domain dependence, as the veracity of FakeHealth articles depends more on factual correctness than thematic divergence. This is also reflected in the near-random (accuracy ~51%) classification results. Generic top-

ics showed weaker performance across datasets, with the ISOT dataset yielding marginally better-than-random results (∼52% accuracy). This suggests that domain-specific topics are more effective at capturing subtle thematic inconsistencies, as generic topics lack the granularity needed for precise detection.

LLM-generated news articles exhibited slightly lower thematic coherence on average compared to human-written articles (0.450 vs. 0.439). However, the differences were minimal, with Cohen's $d = 0.061$ reflecting a very small effect size. Classification results were similarly modest, with threshold classification achieving an accuracy of 53.68%. LLM-generated texts were also found to be more consistent in their thematic structure, reflected in lower standard deviations and reduced peak ratios. Prompting strategies were found to affect thematic coherence, with more constrained prompts (e.g., specified or topical) leading to greater thematic coherence compared to free-form continuation.

Additionally, the explanations generated by the model provided valuable insights into the behavior of the method. These insights, although not a core quantitative result, help in understanding model behavior and inform future refinements.

Concluding, classification based solely on thematic coherence has proven a difficult task. Despite this, the coherence scores provide relevant and interesting insights into the differences between human-written real, human-written fake and LLM-generated news.

6.2 Answering research questions

We will now answer the research questions proposed in section 1.3.

1. What is a method that can effectively measure thematic coherence in news articles?

   To answer this question, we designed an interpretable method that automatically extracts the thematic coherence from news articles. The method leverages an existing state-of-the-art topic model to quantify how themes are discussed throughout the chunks of a text in relation to the full text. We evaluated the method on two coherence evaluation tasks, and it performed well in distinguishing thematically coherent from incoherent texts. This demonstrated the method can effectively measure thematic coherence.

2. What is a suitable evaluation task for capturing thematic coherence?

   A suitable evaluation task for capturing thematic coherence is the sentence replacement task, which introduces thematic disruptions by replacing original sentences with thematically unrelated ones. We first evaluated the method using the sentence ordering task. While traditionally used for evaluating text coherence, it was found to be less informative for assessing thematic coherence, as it focuses more on local sentence transitions

than on the thematic structure. In contrast, the sentence replacement task specifically targets thematic coherence by testing the model's ability to distinguish between thematically coherent and incoherent texts. This allows for a better assessment of disruptions in the thematic structure of a text.

3. How accurately can the proposed coherence method distinguish between human-written real and fake news articles?

   The proposed coherence method can successfully identify significant thematic differences between real and fake news in the ISOT dataset, where real news demonstrates consistently higher coherence. Classification on ISOT shows the method's potential as a valuable additional feature for detecting fake news, but it can best be combined with complementary (local coherence) features for accurate detection. The performance on the FakeHealth dataset and the use of generic topics do highlight the limited utility of the method in narrower domains, where thematic variance is minimal, real and fake articles share closely related topics and veracity distinctions depend on more nuanced factual consistencies.

4. How accurately can the proposed coherence method distinguish between human-written and LLM-generated news articles?

   LLM-generated news articles show slightly lower thematic coherence than human-written ones, with statistically significant but minimal differences. The classification results are modest, indicating that thematic coherence is not sufficient on its own to accurately distinguish between human-written and LLM-generated text. These findings suggest that thematic coherence could be a useful supplementary feature but should be part of a more comprehensive detection approach.

6.3 Limitations and future research

Although the generated insights provide valuable information, some limitations related to the presented findings will be considered. They will be grouped by limitations regarding the method, the topic model and the experiments and results.

6.3.1 Method

It is crucial to reflect on the proposed method and consider certain choices made throughout the process that have influenced this research. The method-related limitations will be discussed in the same order in which the procedure has been discussed before.

First, we will address some limitations related to splitting the articles into chunks. There

are several considerations to be made in determining how to form the chunks. The main parameters that determine this are the minimum number of chunks and the desired number of sentences per chunk. In this research, both have been fixed. We have explored other possibilities and found that a lower number of minimum chunks and a larger number of sentences per chunk resulted in higher thematic coherence on average. This outcome is likely because increasing the input length provides the model with richer context and more information, reducing uncertainty in its topic assignments and, in turn, contributing to improved coherence predictions. Tracking the progression of the discussed themes over multiple segments of a text has to the best of our knowledge not been before in this context and was therefore prioritized in this research. Additional research is required in order to determine the optimal parameters of the model, which could differ per domain. This is left to future work, but supported by the model which is designed to be modular and parametrizable. Another limitation related to chunking is that it occasionally happens that a chunk consists of a couple of shorter sentences, leading to less meaningful topics being assigned to that chunk (like chunk 3 in Figure 5.7 for instance). This could potentially have been solved by creating chunks based on character- or word count, but this would lead to sentences being broken up halfway and was considered more detrimental to the outcomes of this research. Another alternative could have been paragraph splitting, as sentences are now chunked without regard for semantic boundaries, and paragraphs tend to follow these more naturally. This would however require a more strict input format and could lead to very divergent chunk sizes again. Finally, fake news is often spread on social media in short messages like tweets. If these contain 1 or 2 sentences, which is not uncommon, appropriate parameters need to be identified for the model to adequately handle these.

Next, we will consider some limitations regarding the divergence measure. When comparing the topic probability distribution of a chunk against that of the full article, there's an inherent interplay between them. This makes it impossible for the divergence to be 1 (maximal), as the full article topic probability distribution always contains some portion of the chunk it's compared against. This can be considered a limitation because the full article topic probability distribution inherently reflects portions of every chunk, preventing maximal divergence (1) even when the chunk is thematically unrelated. As a result, the measure may underestimate the degree of incoherence in extreme cases. A possible alternative to this divergence computation is to use a pairwise chunk comparison. In future work, we intend to explore this and other possible ways to compare the chunks in the compute divergence module.

Related to the topic probability distribution comparison, in the proposed method all topics are treated equally. We have however observed, for instance in the ISOT dataset, that topics can be semantically similar. This can lead to inflated divergence measures as all topics are

treated as equally distinct while they may actually be somewhat similar. Therefore, in future research it would be highly valuable to incorporate this topic similarity and downweigh divergence values for semantically similar topics. Finally, using thematic divergence between the chunks and the full article does not capture all aspects of the concept coherence. This is also demonstrated by the results on the sentence ordering task, where a human evaluator would most likely consider the permuted articles to be 'incoherent' but the model fails to detect this. This shows that thematic coherence alone does not suffice to capture coherence and should be enhanced with more local, sentence-to-sentence logic like tracking entity transitions in future research.

6.3.2 Topic model

Significant advancements have been made in the field of topic modeling, with traditional approaches like LDA evolving to more complex techniques like BERTopic. Nonetheless, our research highlighted some limitations of this state-of-the-art topic modeling technique. Given that topic modeling has a rather central role in the proposed method, they are relevant to consider. First, topic modeling methods are originally designed to discover latent themes in large collections of documents. When the input text is shorter, which is the case in this research, contextual information may be limited, which results in topics becoming less meaningful. This was also demonstrated by the example explanation included in section 5.4.

Another caveat of BERTopic relying on density-based clustering is that when it assigns a relatively large portion of a document to the outlier category, BERTopic is very inclined to pick the most common topic in the dataset as the next-best. This may influence the coherence scores in two ways. If this happens for only a single chunk and the remainder of the chunks are very aligned with the full article topics, it leads to a single high divergence value, disproportionally increasing the observed incoherence. However if the uncertainty is high for both the full article and the majority of its chunks, it may disproportionally consider the text to be aligned even though semantically it may not be so consistent. In future research, these outliers should be more closely integrated, for instance by weighing down or excluding divergence values if the extracted outlier probability exceeds a certain threshold. This should mitigate the bias towards assigning the most common topics to a text.

Finally, the BERTopic pipeline contains many different options in terms of modules and parameters. Optimizing the perfect configuration was not the scope of this research, but given the numerous possibilities, careful tuning of all hyperparameters could potentially lead to better topic representations.

6.3.3 Experiments and results

Finally, some limitations related to the conducted experiments and the results can be discerned. The first potential limitation related to the experiments is the simplicity of the chosen classifiers. These relatively straightforward classification methods were opted for to ensure interpretability in its verdicts. Some classifiers, such as an SVM or a random forest, were tested and offered limited additional insights and were therefore deliberately not included. If one's objective is to pursue the maximal classification score whilst retaining interpretability, neural networks together with model-specific or -agnostic explainability techniques could be considered in the future. However, this was not opted for as it can quickly become complex, make thorough dissection of thematic coherence more difficult and draw away from the focus of this research.

A second limitation is the generalization of the findings to new domains. The findings suggest thematic patterns may vary across different topic areas (and this is not even considering different languages yet). Ensuring good generalization, however, also requires high-quality and diverse data. For fake news detection, data labeling is time-consuming and -critical due to the constant emergence of new false information. Similarly, for the detection of LLM-generated text, high-quality human-written texts from various domains are required that should be prevented from leaking into the training data of LLMs, which is in practice hard to solve. This highlights the difficulties related to high-quality data that may also influence good generalization of the results.

Another factor that may affect the obtained results is the general focus on only the RMSE. While the RMSE encapsulates the divergence values and contains information on both the magnitude of deviations and the variability across chunks, exploring alternative ways to combine divergence scores could provide more nuanced measures. For instance, a weighted aggregation measure that penalizes highly scattered distributions could mitigate cases where the full article topic distribution aligns artificially well with incoherent chunks due to scatter. Additionally, while this thesis focuses on coherence-based measures, integrating these metrics into broader fake news detection frameworks is an exciting avenue for future research. For example, combining coherence metrics with language-based features that have been proven indicative of fake news (such as emotional extremes or misleading headlines) could potentially improve classification accuracy. Enhancing this with fact-based features would further enrich the detection framework and provide the most comprehensive way of detecting fake news, but this is beyond the scope of this thesis.

6.4 Conclusion

In this thesis, a novel method of capturing thematic coherence has been proposed. This is done by measuring the progression of topics across chunks of an article and comparing them to the overall topic(s) discussed in the full article. We also introduced a new task, the sentence replacement task, for measuring the thematic coherence of text. This extends the pool of tasks used in the literature to measure overall text coherence. The sentence replacement task consists of replacing sentences with thematically unrelated ones to create thematically incoherent articles. The proposed method is effective at capturing these thematically incoherent from coherent texts. Both the proposed method and the newly designed sentence replacement task are research contributions to the existing literature. In the context of fake news detection, the proposed method has shown its ability to capture meaningful differences in thematic coherence between real and fake articles. While its standalone predictive power in detecting fake news is modest, the method has demonstrated a clear distinctive capability, making it a valuable addition to state-of-the-art models that analyze the language of fake news. Similarly, its potential extends to style-based detectors for LLM-generated text where thematic coherence can serve as an additional feature. The importance of further research in this domain is substantial: a comprehensive understanding of the characteristics of fake news and LLM-generated news becomes increasingly vital in an age where the boundaries of truthfulness are constantly challenged.

Bibliography

[1] R. Garrett, "Echo chambers online?: Politically motivated selective exposure among internet news users," *Journal of computer-mediated communication*, vol. 14, no. 2, pp. 265–285, 2009.

[2] E. Parliament, 2023, Accessed 5 August 2024. [Online]. Available: `https://www.europarl.europa.eu/news/en/press-room/20231115IPR11303/tv-still-main-source-for-news-but-social-media-is-gaining-ground`.

[3] M. Flintham, C. Karner, K. Bachour, H. Creswick, N. Gupta, and S. Moran, "Falling for fake news: Investigating the consumption of news via social media," *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, pp. 1–10, 2018.

[4] X. Zhang and A. A. Ghorbani, "An overview of online fake news: Characterization, detection, and discussion," *Information Processing and Management*, vol. 52, no. 2, 2020.

[5] G. Pennycook, T. D. Cannon, and D. G. Rand, "Prior exposure increases perceived accuracy of fake news," *Journal of experimental psychology: general*, vol. 147, no. 12, 2018.

[6] B. Horne and S. Adali, "This just in: Fake news packs a lot in title, uses simpler, repetitive content in text body, more similar to satire than real news," *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 11, no. 1, pp. 759–766, 2017.

[7] A. Bovet and H. A. Makse, "Influence of fake news in twitter during the 2016 us presidential election," *Nature communications*, vol. 10, no. 1, 2019.

[8] B. Riedel, I. Augenstein, G. P. Spithourakis, and S. Riedel, "A simple but tough-to-beat baseline for the fake news challenge stance detection task," 2017.

[9] N. Grinberg, K. Joseph, L. Friedland, B. Swire-Thompson, and D. Lazer, "Fake news on twitter during the 2016 us presidential election," *Science*, vol. 363, no. 6425, pp. 374–378, 2019.

[10] X. Zhou and R. Zafarani, "A survey of fake news: Fundamental theories, detection methods, and opportunities," *ACM Computing Surveys (CSUR)*, vol. 53, no. 5, pp. 1–40, 2020.

[11] L. Hadlington, L. J. Harkin, D. Kuss, K. Newman, and F. C. Ryding, "Perceptions of fake news, misinformation, and disinformation amid the covid-19 pandemic: A qualitative exploration," *Psychology of Popular Media*, vol. 12, no. 1, 2023.

[12] K. Shu, A. Sliva, S. Wang, J. Tang, and H. Liu, "Fake news detection on social media: A data mining perspective," *ACM SIGKDD explorations newsletter*, vol. 19, no. 1, pp. 22–36, 2017.

[13] L. V. Lakshmanan, M. Simpson, and S. Thirumuruganathan, "Combating fake news: A data management and mining perspective," *Proceedings of the VLDB Endowment*, vol. 12, no. 12, pp. 1990–1993, 2019.

[14] S. Vosoughi, D. Roy, and S. Aral, "The spread of true and false news online," *International Journal of Machine Learning and Cybernetics*, vol. 359, no. 6380, pp. 1146–1151, 2018.

[15] J. Paschen, "Investigating the emotional appeal of fake news using artificial intelligence and human contributions," *Journal of Product and Brand Management*, vol. 29, no. 2, pp. 223–233, 2020.

[16] S. Feng, R. Banerjee, and Y. Choi, "Syntactic stylometry for deception detection.," *In Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, vol. 2, pp. 171–175, 2012.

[17] C. Castillo, M. Mendoza, and B. Poblete, "Information credibility on twitter.," *In Proceedings of the 20th international conference on World wide web*, pp. 675–684, 2011.

[18] K. Sharma, F. Qian, H. Jiang, N. Ruchansky, M. Zhang, and Y. Liu, "Combating fake news: A survey on identification and mitigation techniques," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 10, no. 3, pp. 1–42, 2019.

[19] M. S. Dogo, P. Deepak, and A. Jurek-Loughrey, "Exploring thematic coherence in fake news," *Proceedings of the ECML PKDD 2020 Workshops: Workshops of the European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 571–580, 2020.

[20] H. Karimi and J. Tang, "Learning hierarchical discourse-level structure for fake news detection," 2019.

[21] V. L. Rubin and T. Lukoianova, "Truth and deception at the rhetorical structure level.," *Journal of the Association for Information Science and Technology*, vol. 66, no. 5, pp. 905–917, 2015.

[22] L. Fröhling and A. Zubiaga, "Feature-based detection of automated language models: Tackling gpt-2, gpt-3 and grover.," *PeerJ Computer Science*, vol. 7, 2021.

[23] K. Hu and K. Hu, "Chatgpt sets record for fastest-growing user base - analyst note," *Reuters*, 2023, Accessed 27 December 2024. [Online]. Available: `https://www.reuters.com/technology/chatgpt-sets-record-fastest-growing-user-base-analyst-note-2023-02-01/`.

[24] S. Badaskar, S. Agarwal, and S. Arora, "Identifying real or fake articles: Towards better language modeling," *In Proceedings of the Third International Joint Conference on Natural Language Processing*, vol. 2, 2008.

[25] A. Maimon and R. Tsarfaty, "Cohesentia: A novel benchmark of incremental versus holistic assessment of coherence in generated texts.," 2023.

[26] M. Madani, H. Motameni, and R. Roshani, "Fake news detection using feature extraction, natural language processing, curriculum learning, and deep learning," *International Journal of Information Technology and Decision Making*, pp. 1–36, 2023.

[27] H. C. Moon, T. Mohiuddin, S. Joty, and X. Chi, "A unified neural coherence model.," 2019.

[28] I. Singh, P. Deepak, and K. Anoop, "On the coherence of fake news articles," *Proceedings of the ECML PKDD 2020 Workshops: Workshops of the European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 591–607, 2020.

[29] R. Barzilay and M. Lapata, "Modeling local coherence: An entity-based approach.," *Computational Linguistics*, vol. 34, no. 1, pp. 1–34, 2008.

[30] C. Guinaudeau and M. Strube, "Graph-based local coherence modeling.," *In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, vol. 1, pp. 93–103, 2013.

[31] M. Mesgar, L. F. Ribeiro, and I. Gurevych, "A neural graph-based local coherence model.," *In Findings of the Association for Computational Linguistics: EMNLP 2021*, pp. 2316–2321, 2021.

[32] R. Tang, Y. N. Chuang, and X. Hu, "The science of detecting llm-generated text.," *Communications of the ACM*, vol. 67, no. 4, pp. 50–59, 2024.

[33] J. Posetti and A. Matthews, "A short guide to the history of 'fake news' and disinformation," *International Center for Journalists*, vol. 7, 2018.

[34] E. Aïmeur, S. Amri, and G. Brassard, "Fake news, disinformation and misinformation in social media: A review," *Social Network Analysis and Mining*, vol. 13, no. 1, 2023.

[35] K. Shu, S. Wang, and H. Liu, "Beyond news contents: The role of social context for fake news detection," *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, pp. 312–320, 2019.

[36] A. Bessi and E. Ferrara, "Social bots distort the 2016 us presidential election online discussion," *First Monday*, vol. 21, 2016.

[37] K. Shu, X. Zhou, S. Wang, R. Zafarani, and H. Liu, "The role of user profiles for fake news detection," *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pp. 436–439, 2019.

[38] Z. Chu, S. Gianvecchio, H. Wang, and S. Jajodia, "Who is tweeting on twitter: Human, bot, or cyborg?" *Proceedings of the 26th Annual Computer Security Applications Conference*, pp. 21–30, 2010.

[39] C. Shao, G. L. Ciampaglia, O. Varol, A. Flammini, and F. Menczer, "The spread of fake news by social bots," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 96, 2017.

[40] S. Kwon, M. Cha, K. Jung, W. Chen, and Y. Wang, "Prominent features of rumor propagation in online social media," *Proceedings of the 2013 IEEE 13th International Conference on Data Mining*, vol. 96, pp. 1103–1108, 2013.

[41] P. H. A. Faustini and T. F. Covoes, "Fake news detection in multiple platforms and languages.," *Expert Systems with Applications*, vol. 158, 2020.

[42] H. Wang, C. Dou Y.and Chen, L. Sun, P. S. Yu, and K. Shu, "Attacking fake news detectors via manipulating news social engagement.," *In Proceedings of the ACM Web Conference 2023*, pp. 3978–3986, 2023.

[43] S. Cresci, R. Di Pietro, M. Petrocchi, A. Spognardi, and M. Tesconi, "Fame for sale: Efficient detection of fake twitter followers," *Decision Support Systems*, vol. 80, pp. 56–71, 2015.

[44] Z. Jin, J. Cao, Y. Zhang, and J. Luo, "News verification by exploiting conflicting social viewpoints in microblogs," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 30, no. 1, 2016.

[45] P. Saikia, K. Gundale, A. Jain, D. Jadeja, H. Patel, and M. Roy, "Modelling social context for fake news detection: A graph neural network based approach," *Proceedings of the 2022 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, 2022.

[46] L. Jacobson, *Politifact*, Accessed: 2024-07-07, 2024. [Online]. Available: `https://www.politifact.com/factchecks/list/?speaker=joe-biden`.

[47] Z. Guo, M. Schlichtkrull, and A. Vlachos, "A survey on automated fact-checking," *Transactions of the Association for Computational Linguistics*, vol. 10, pp. 178–206, 2022.

[48] J. Thorne, A. Vlachos, C. Christodoulopoulos, and A. Mittal, "Fever: A large-scale dataset for fact extraction and verification," 2018.
[49] X. Zeng, A. S. Abumansour, and A. Zubiaga, "Automated fact-checking: A survey," *Language and Linguistics Compass*, vol. 15, no. 10, 2021.
[50] J. Hirschberg and C. D. Manning, "Advances in natural language processing.," *Science*, vol. 349, no. 6245, pp. 261–266, 2015.
[51] S. Afroz, M. Brennan, and R. Greenstadt, "Detecting hoaxes, frauds, and deception in writing style online," *Proceedings of the 2012 IEEE Symposium on Security and Privacy*, pp. 461–475, 2012.
[52] Y. Zhang, R. Jin, and Z. H. Zhou, "Understanding bag-of-words model: A statistical framework," *International Journal of Machine Learning and Cybernetics*, vol. 1, pp. 43–52, 2010.
[53] I. A. Sag, T. Wasow, E. M. Bender, and I. A. Sag, "Syntactic theory: A formal introduction," *Stanford, CA: Center for the Study of Language and Information.*, vol. 92, 1999.
[54] M. K. Balwant, "Bidirectional lstm based on pos tags and cnn architecture for fake news detection.," *In 2019 10th International conference on computing, communication and networking technologies (ICCCNT)*, pp. 1–6, 2019.
[55] P. Baroni, M. Caminada, and M. Giacomin, "An introduction to argumentation semantics.," *The knowledge engineering review*, vol. 26, no. 4, pp. 365–410, 2011.
[56] R. K. Kaliyar, A. Goswami, and P. Narang, "Fakebert: Fake news detection in social media with a bert-based deep learning approach.," *Multimedia tools and applications*, vol. 80, no. 8, pp. 11 765–11 788, 2021.
[57] L. Wu and Y. Rao, "Adaptive interaction fusion networks for fake news detection.," *In ECAI*, pp. 2220–2227, 2020.
[58] C. Guo, J. Cao, X. Zhang, K. Shu, and M. Yu, "Exploiting emotions for fake news detection on social media.," 2019.
[59] X. Zhou, A. Jain, V. V. Phoha, and R. Zafarani, "Fake news early detection: A theory-driven model.," *Digital Threats: Research and Practice*, vol. 1, no. 2, pp. 1–25, 2019.
[60] J. Kapusta, Ľ. Benko, and M. Munk, "Fake news identification based on sentiment and frequency analysis," *Proceedings of the EMENA-ISTL*, vol. 3, pp. 400–409, 2020.
[61] e. a. Su Jinyan, "Fake news detectors are biased against texts generated by large language models.," 2023.
[62] M. Ott, Y. Choi, C. Cardie, and J. T. Hancock, "Finding deceptive opinion spam by any stretch of the imagination.," 2011.
[63] B. Bamberg, "What makes a text coherent?" *College Composition and Communication*, vol. 34, no. 4, pp. 417–429, 1983.
[64] C. Dictionary, 2024, Accessed 15 July 2024. [Online]. Available: `https://dictionary.cambridge.org/dictionary/english/coherence`.
[65] I. Lee, "Teaching coherence to esl students: A classroom inquiry," *Journal of Second Language Writing*, vol. 11, no. 2, pp. 135–159, 2002.
[66] I. Bruce, "Towards an eap without borders: Developing knowledge, practitioners, and communities," *International Journal of English for Academic Purposes: Research and Practice*, pp. 23–36, 2021.
[67] W. C. Mann and S. A. Thompson, "Rhetorical structure theory: Toward a functional theory of text organization," *Text-interdisciplinary Journal for the Study of Discourse*, vol. 8, no. 3, pp. 243–281, 1988.
[68] N. M. Anspach, J. T. Jennings, and K. Arceneaux, "A little bit of knowledge: Facebook's news feed and self-perceptions of knowledge," *Research and Politics*, vol. 6, no. 1, 2019.
[69] W. Ferreira and A. Vlachos, "Emergent: A novel data-set for stance classification," *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2016.
[70] S. Yoon, K. Park, J. Shin, *et al.*, "Detecting incongruity between news headline and body text via a deep hierarchical encoder," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 1, pp. 791–800, 2019.
[71] J. Li and E. Hovy, "A model of coherence based on distributed sentence representation," *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 2039–2048, 2014.
[72] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, 2014.

[73] M. Norouzi, T. Mikolov, S. Bengio, *et al.*, "Zero-shot learning by convex combination of semantic embeddings," 2013.

[74] E. Gabrilovich and S. Markovitch, "Computing semantic relatedness using wikipedia-based explicit semantic analysis," *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, vol. 7, pp. 1606–1611, 2007.

[75] J. Hoffart, M. A. Yosef, I. Bordino, *et al.*, "Robust disambiguation of named entities in text," *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pp. 782–792, 2011.

[76] M. Casillo, F. Colace, B. B. Gupta, D. Santaniello, and C. Valentino, "Fake news detection using lda topic modelling and k-nearest neighbor classifier.," *In Computational Data and Social Networks: 10th International Conference, CSoNet 2021, Virtual Event*, vol. 10, pp. 330–339, 2021.

[77] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation.," *Journal of machine Learning research*, vol. 3, pp. 993–1022, 2003.

[78] S. V. Raju, B. K. Bolla, D. K. Nayak, and J. Kh, "Topic modelling on consumer financial protection bureau data: An approach using bert based embeddings.," *In 2022 IEEE 7th International conference for Convergence in Technology (I2CT)*, pp. 1–6, 2022.

[79] M. Grootendorst, 2020, Accessed 18 December 2024. [Online]. Available: `https://maartengr.github.io/BERTopic/index.html`.

[80] J. Ramos, "Using tf-idf to determine word relevance in document queries.," *In Proceedings of the first instructional conference on machine learning*, vol. 42, no. 1, pp. 29–48, 2002.

[81] A. C. Graesser, D. S. McNamara, M. M. Louwerse, and Z. Cai, "Coh-metrix: Analysis of text on cohesion and language.," *Behavior research methods, instruments, and computers*, vol. 36, no. 2, pp. 193–202, 2004.

[82] M. Mesgar and M. Strube, "Lexical coherence graph modeling using word embeddings.," *In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1414–1423, 2016.

[83] S. E. Schwarm and M. Ostendorf, "Reading level assessment using support vector machines and statistical language models.," *In Proceedings of the 43rd annual meeting of the Association for Computational Linguistics (ACL'05)*, pp. 523–530, 2005.

[84] D. T. Nguyen and S. Joty, "A neural local coherence model.," *In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, vol. 1, pp. 1320–1330, 2017.

[85] M. Schlichtkrull, T. N. Kipf, P. Bloem, R. Van Den Berg, I. Titov, and M. Welling, "Modeling relational data with graph convolutional networks.," *The semantic web: 15th international conference, ESWC 2018, Heraklion, Crete, Greece, June 3–7, 2018, proceedings 15.*, pp. 593–607, 2018.

[86] T. Mohiuddin, S. Joty, and D. T. Nguyen, "Coherence modeling of asynchronous conversations: A neural entity grid approach.," 2018.

[87] Y. Huang, L. Sun, H. Wang, S. Wu, Z. Q., and Y. Li Y. andZhao, "Trustllm: Trustworthiness in large language models.," 2024.

[88] E. Clark, T. August, S. Serrano, N. Haduong, S. Gururangan, and N. A. Smith, "All that's' human'is not gold: Evaluating human evaluation of generated text.," 2021.

[89] A. Uchendu, Z. Ma, T. Le, R. Zhang, and D. Lee, "Turingbench: A benchmark environment for turing test in the age of neural text generation.," 2021.

[90] V. S. Sadasivan, A. Kumar, S. Balasubramanian, W. Wang, and S. Feizi, "Can ai-generated text be reliably detected?," 2022.

[91] I. Vayansky and S. A. Kumar, "A review of topic modeling methods.," *Information Systems*, vol. 94, no. 101582, 2020.

[92] M. Mohammadi, N. Ali Khan, H. Hassanpour, and A. Hussien Mohammed, "Spike detection based on the adaptive time–frequency analysis.," *Circuits, Systems, and Signal Processing*, vol. 39, pp. 5656–5680, 2020.

[93] L. H. Gilpin, D. Bau, B. Z. Yuan, A. Bajwa, M. Specter, and L. Kagal, "Explaining explanations: An overview of interpretability of machine learning.," *In 2018 IEEE 5th International Conference on data science and advanced analytics (DSAA)*, pp. 80–89, 2018.

[94] E. L. Lehmann, J. P. Romano, and G. Casella, "Testing statistical hypotheses," vol. 3, 1986.

[95] H. Ahmed, I. Traore, and S. Saad, "Detection of online fake news using n-gram analysis and machine learning techniques.," *In Intelligent, Secure, and Dependable Systems in Distributed and Cloud Environments*, vol. 1, pp. 127–138, 2017.

[96] E. Dai, Y. Sun, and S. Wang, "Ginger cannot cure cancer: Battling fake health news with a comprehensive data repository.," *In Proceedings of the International AAAI Conference on Web and Social Media*, vol. 14, pp. 853–862, 2020.

[97] K. Shu, D. Mahudeswaran, S. Wang, D. Lee, and H. Liu, "Fakenewsnet: A data repository with news content, social context, and spatiotemporal information for studying fake news on social media.," *Big data*, vol. 8, no. 3, pp. 171–188, 2020.

[98] A. Abdelrazek, Y. Eid, E. Gawish, W. Medhat, and A. Hassan, "Topic modeling algorithms and applications: A survey.," *Information systems*, vol. 112, 2023.

[99] K. H. Huang, K. McKeown, P. Nakov, Y. Choi, and H. Ji, "Faking fake news for real fake news detection: Propaganda-loaded training data generation.," 2022.

[100] Y. Li, Q. Li, L. Cui, *et al.*, "Mage: Machine-generated text detection in the wild.," *In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, vol. 1, pp. 36–53, 2024.

A. Appendix

Topic ID	Topic Label	Count
-1	Outlier	720
0	Corporate Earnings & Revenue	206
1	Soviet Union & Gorbachev Politics	85
2	Television Broadcasting & Networks	80
3	Fiscal Policies & Government Actions	69
4	Legal Proceedings & Prosecution	52
5	Corporate Executives & Agencies	50
6	Oil Refining & Exxon Mobil	45
7	Treasury & Bond Markets	45
8	Economic Recession & Inflation	43
9	FDA & Pharmaceutical Developments	42
10	Earthquakes & Natural Disasters	41
11	UAL Takeovers & Stock Movements	39
12	Securities & Shareholder Activities	39
13	Stock Market Volatility & Dow Jones	36
14	Automotive Takeovers & Shareholders	35
15	Banking & Financial Institutions	31
16	Magazine Publishing & Media	29
17	Stock Trading & Market Participants	27
18	Bank Loans & Collateral Policies	25
19	Japan & Tokyo-based Corporations	22
20	Telecommunications & Earnings	21
21	Japanese Yen & Sales Revenue	19
22	Nikkei Index & Japanese Stock Markets	18
23	Treasury Bonds & Merrill Lynch	17

Table 1: WSJ dataset topic overview.

Topic ID	Topic Label	Count
-1	Outlier	10678
0	Trump, Mueller Investigation & Russia	4062
1	US Politics: McConnell, Cruz & Nominees	1956
2	Tax Reform and Congressional Debates	1502
3	Israeli-Palestinian Conflict	1334
4	NFL Protests and Anthem Controversy	1015
5	Gun Control & Firearm Legislation	955
6	Immigration Policies and Deportation	931
7	Kurdish Issues in Iraq & Syria	923
8	North Korea: Pyongyang & Missiles	769
9	Obamacare & Healthcare Repeal	652
10	LGBT Rights and Abortion Policies	573
11	Rohingya Refugees in Myanmar	566
12	Iran Nuclear Sanctions & Diplomacy	528
13	Puerto Rico: Hurricane Recovery	435
14	Catalonia Independence & Spain Politics	431
15	Venezuelan Politics & Chavez Legacy	428
16	Middle-East Militants & Troop Activity	380
17	German Politics: Merkel & Coalitions	326
18	Brexit Negotiations & EU Relations	324
19	Russia: Putin & Sanctions	310
20	Zimbabwe Politics: Mugabe's Leadership	310
21	Terrorism, Suspects & Countermeasures	279
22	Illinois Budget & Fiscal Policies	273
23	Refugees & Migration Challenges	243
24	FCC & Net Neutrality Debate	212
25	Turkey Politics: Erdogan's Policies	171
26	Chinese Politics: Mao & Zedong Legacy	156
27	Media: Radio & Broadcasts	152
28	Earthquakes & Disaster Relief	141
29	US Elections: Giuliani & Romney	137
30	Philippines Politics: Duterte's Leadership	130
31	Congress Allegations & Misconduct	107
32	Flint Water Crisis & Contamination	105
33	New Zealand: Elections & Politics	104
34	Japan Politics: Abe & Elections	89
35	Hastert Trial: Sentencing & Scandals	61
36	Bosnia & Serbian Ethnic Conflict	61
37	Pakistan Politics: Sharif Leadership	58
38	Canada Politics: Trudeau & Leadership	57
39	Activism: Soros & Koch Influence	49
40	Bridgegate Scandal: Christie	46
41	Marijuana Legalization & Cannabis	43
42	Bali Volcanic Eruptions	40
43	Saudi Arabia: Women's Rights Reform	39
44	JavaScript in Web Contexts	39
45	Food & Pie-Related Topics	26
46	Naval Issues: Submarines & Sailors	25
47	British Royals: Prince Harry & Markle	22
48	Rock Musicians & Pop Culture	17

Table 2: ISOT dataset topic overview.

Topic ID	Topic Label	Count
-1	Outlier	5315
0	Armenian Genocide and History	1429
1	Christianity and Religious Beliefs	791
2	Space Exploration and Astronomy	686
3	Computer Hardware and Software	633
4	Political Discussions	492
5	Car Maintenance and Performance	462
6	Gun Ownership and Usage	438
7	Healthcare and Medical Innovations	410
8	Atheism and Religious Critique	379
9	Sports - Hockey	347
10	Cryptography and Network Security	298
11	Video Games and Entertainment	278
12	Electronics and Hardware Design	260
13	Middle East Politics	255
14	Operating Systems - Windows	253
15	Ethics and Philosophy	250
16	Computer Graphics and Visualization	227
17	Politics - U.S.	194
18	Legal Issues and Law Enforcement	181
19	Sports - Baseball	178
20	Technology News and Innovations	155
21	Movies and Media	128
22	Fitness and Nutrition	84
23	Education and Learning	83
24	Programming and Coding	82
25	Space Missions and Satellites	74
26	Networking and Internet Technologies	70
27	Jobs and Career Advice	63
28	Psychology and Human Behavior	63
29	Hardware Issues and Repairs	44
30	Environment and Climate Change	43
31	AI and Machine Learning	41
32	Society and Culture	37
33	Travel and Tourism	36
34	Parenting and Family Life	32
35	Historical Events and Analysis	31
36	Scientific Research and Discoveries	30
37	Music and Artists	30
38	Books and Literature	28
39	Space Science and Theories	28
40	Economics and Business	27
41	Sports - Basketball	25
42	Photography and Art	24
43	Gaming Communities	24
44	Technology Reviews and Tips	21
45	Religion and Spirituality	20
46	Debates and Controversies	19
47	Military and Warfare	17
48	Humor and Memes	16

Table 3: 20 Newsgroups dataset topic overview.