Research Master Thesis

Diversifying News Recommendation Systems by Detecting Fragmentation in News Story Chains

Alessandra Polimeno

a thesis submitted in partial fulfilment of the requirements for the degree of

MA Linguistics
(Human Language Technology)

Vrije Universiteit Amsterdam

Computational Lexicology and Terminology Lab
Department of Language and Communication
Faculty of Humanities

Supervised by: prof. dr. Antske Fokkens, Myrthe Reuver, and Sanne Vrijenhoek
2^{nd} reader: dr. Lisa Beinborn

Submitted: July 1, 2022

Abstract

This thesis contributes to a line of research that aims to develop measures for diversity in the context of personalized news recommendation systems. The focus lies on the Fragmentation metric, which measures the overlap in news story chains that users are exposed to in their personalized news recommendations. News story chains consist of articles that report on the same action that took place at a specific time. A low Fragmentation Score indicates that readers are exposed to the same chains, possibly from different perspectives. This implies the existence of a common public sphere in which people are aware of the same events that are happening in society. On the other hand, a high Fragmentation Score indicates the existence of specialized bubbles in which there is a discrepancy in the events that people read about. Previous work on the Fragmentation metric has used clustering to group articles into news story chains, but an extensive evaluation of the clustering performance has not yet been done.

The contribution of this thesis is two-fold. Firstly, the performance of various text representation methods and clustering algorithms are compared on the task of news story chain detection. Secondly, the effect of errors in the news chain detection pipeline on the resulting Fragmentation Score are investigated. This was done by comparing the Fragmentation Scores that each setup generates based on three scenarios that simulate news recommendations that display a varying degree of Fragmentation. To my knowledge, this thesis is the first project to develop a pipeline that systematically evaluates a combination of systems on the task of news story chain detection, as well as reporting the resulting Fragmentation Score.

The results indicate that the implementation of a text representation method that is specialized in capturing semantic similarity is a prerequisite for a high performance on the task of news story chain detection. More specifically, Sentence-BERT sentence embeddings combined with the agglomerative hierarchical clustering algorithm was found to outperform both the baseline and the other experimental setups. Moreover, the setups that achieved the highest performance on the task of news story chain detection were found to result in the most accurate Fragmentation Score across the three scenarios.

Declaration of Authorship

I, Alessandra Polimeno, declare that this thesis, titled *Diversifying News Recommendation Systems by Detecting Fragmentation in News Story Chains* and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Date: 01/07/2022

Signed:

Acknowledgments

Firstly, I want to express my gratitude to prof. dr. Antske Fokkens for her supervision of this thesis project. Her guidance and feedback allowed me to gain more insight into the workings of clustering, and supported me in the process of writing a thesis I can be proud of. Secondly, I want to thank Sanne Vrijenhoek and Myrthe Reuvers for the additional supervision. Sanne's expertise on the diversity metrics provided me with the context that was necessary to perform the experiments. Myrthe's knowledge of NLP, and her inventive ideas often helped me out when I did not know how to proceed. Lastly, a big thank you to Charlotte Pouw and Eliza Hobo for being pillars of support. They helped me to stay motivated throughout the lockdowns, and added much more fun to my time at the VU.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

In November 2019, an interdisciplinary group of researchers from the fields of psychology, sociology, economics, law, and computer science came together to talk about the need for an increase in diversity in personalized news recommendation systems (NRSs), and collected their conclusions in a manifesto (see Bernstein et al. (2019)). They state that our news intake is increasingly more often curated by algorithmic recommendations, which calls attention to the responsibility such systems have in steering our reading behaviour. As recommendation systems select the information that readers are exposed to, they "take on a powerful gatekeeping function in the information ecosystem" (Bernstein et al., 2019, p. 47). One of the recommendations that is stated in the manifesto is that "new measures and models of diversity are needed as current models of diversity, typically, do not capture the multidimensionality of diversity" (Bernstein et al., 2019, p. 56). In other words, the evaluation of news recommendation systems should include measures that follow a more in-depth definition of diversity.

Depending on the perspective of the media organization, a diverse and democratic NRS should expose readers to a variety of topics and perspectives that enable them to form informed opinions. Alternatively, NRSs may play a role in breaking filter bubbles (i.e. situations in which online users continuously encounter information that confirm and reinforce their own current beliefs) by recommending articles that fall outside the pool of articles that a reader usually reads, but might still be interesting to them. Others may argue that a NRS should help the user to learn more about the topics they are already interested in. Vrijenhoek et al. (2021) argue that metrics of diversity should be able to quantify these expectations. In contrast, traditional evaluation metrics of NRSs generally value short-term engagement by maximizing clicks: a system is deemed accurate if it recommends articles that receive many clicks (McNee et al., 2006). The use of clickbait (i.e. often sensationalized titles or messages that aim to lure readers to click on an article) is commonplace but comes at the cost of being uninformative or even deceiving (Scott, 2021). Moreover, evaluating a NRS in terms of short-term engagement leads to homogeneous recommendations and it endorses the existence of filter bubbles (McNee et al., 2006).

Based on the work by Helberger (2019), a set of five diversity metrics was developed by Vrijenhoek et al. (2021). It is a first attempt at quantifying diversity in the context of NRSs. The focus of the project was on the conceptual underpinnings of such metrics. The proposed metrics are analyzed in the light of theories of democracy, where different theories prescribe different ideal democratic values. The main goal of the current thesis is to expand and investigate the technical implementation of one such metrics, namely the *Fragmentation Score*. Fragmentation refers to the degree to which a group of users is exposed to the same *news story chains*. News story chains consist of articles that report on the same action that took place at a specific time and place (Nicholls and Bright, 2019). They are important to consider since media outlets do not always publish topically independent articles, but rather tend to report follow-up items on ongoing events, resulting in chains. The Fragmentation Score thus calculates the overlap in story chains between the recommendation sets of users.

This thesis shifts the focus from a conceptual development of the Fragmentation metric to a technical implementation by testing how various clustering methods perform at the task of news story chain detection, and how this performance subsequently affects the resulting Fragmentation Score. The following research questions are addressed:

1. How do various clustering approaches perform on the task of news story chain detection?
2. How is the Fragmentation Score influenced by variations in the chain detection system?

Before clustering can be performed, the article texts should be transformed into machine-readable representations. These representations will have a large impact of the ability of a clustering algorithm to group articles into news story chains. This, in turn, has effect on the Fragmentation Score that follows. For this reason, the following subquestion is addressed:

– How do different representations of news articles influence both the chain detection and Fragmentation Score?

A pipeline with four components was developed to investigate the research questions. The first step consists of obtaining the representations of the articles, for which the following methods are used: Bag of Words, word embeddings with GloVe Pennington et al. (2014), and sentence embeddings with Sentence-BERT Reimers and Gurevych (2019). In the second step, the articles are clustered into news story chains by means of agglomerative hierarchical clustering and the DB-Scan algorithm. The resulting clusters are intrinsically evaluated by a number of evaluation metrics, of which the V-measure is the most important. The third step is concerned with generating sets of news recommendations for simulated users, as this is required to measure Fragmentation. Three scenarios are developed where the number of news story chains that readers are exposed to varies, which leads to different Fragmentation Scores. The readers receive one article from each story chain in Scenario 1, leading to a low Fragmentation Score. In Scenario 2, they are exposed to articles from a single chain, which results in a high Fragmentation Score. Scenario 3 represents more realistic reading behavior, as it constructs three distinct profiles with different preferences in terms of the number of story chains they are exposed to. This results in a more balanced Fragmentation Score. The recommendation provide 7 articles per scenario for 1000 users. In the final step, the Fragmentation Score is calculated over each scenario by using the predicted story chains that the systems resulting from the previous step generated. Extrinsic evaluation is done by means of comparing the resulting Fragmentation Score of the various clustering setups. This allows for the investigation of the effect of errors in each pipeline component on the resulting Fragmentation Score.

We found that the model that combines sentence embeddings with agglomerative hierarchical clustering performs best on the task of news story chain detection. The second-best performing system is sentence embeddings combined with the DB-Scan algorithm. This highlights the importance of the use of a text representation method that is specialized at encoding semantic information for this task. The Fragmentation Score that results from this system was found to reflect the expectations for each scenario. Moreover, it was found that systems that perform poorly at news chain detection result in a Fragmentation Score that exhibits low variability across scenarios. Measuring the Fragmentation Score over various scenarios and comparing the variability in the resulting scores can thus be a useful method for initial cluster validation.

The structure of this thesis is as follows: Chapter 2 provides background information on a variety of topics and terminology that is relevant in the current context, as well as an overview of previous approaches to news story chain detection. In Chapter 3, the HeadLine Grouping Dataset is described, which is used in this project because it contains articles with annotated news story chains. The experimental setup is outlined in Chapter 4, followed by the results and analysis of both experiments in Chapter 5. Finally, the discussion and conclusion can be found in Chapter 6.

Chapter 2

Background and Related Work

This chapter provides the background information that is needed to understand this thesis. It starts with a brief introduction into the workings of news recommendation systems, where a typology of different types of NRSs is discussed, as well as the user data that is needed to generate recommendations (Section 2.1). Secondly, Section 2.2 describes the context in which the Fragmentation metric is developed, namely the necessity for diversity metrics that align with different conceptualizations of democracy. Then, the concept of news story chains is defined more precisely, and previous approaches to the task of detecting the chains is reported (Section 2.3). The remaining sections provide information on the steps that should be taken when building a pipeline that aims to detect news story chains, namely text representation (Section 2.4), clustering algorithms (Section 2.5), and clustering validation (Section 2.6).

2.1 Overview of News Recommendation Systems

Personalized news recommendation systems aim to provide users with a selected news feed that contains the content they are predicted to find interesting. The ever-expanding source of information that the internet provides can cause users to be overwhelmed, making the process of finding the content they find interesting and important time-consuming. To address this problem, automatic recommendation systems have been developed to make selections such that the user can find content in a reasonable amount of time (Kunaver and Požrl, 2017). The task of recommendation is generally defined as a ranking problem, where a collection is selected from a pool of items, and ranked in a way that increases the chance that the user will find the items they are most likely to appreciate at the top (Li and Wang, 2019).

In the context of news recommendation, most recommenders form their predictions on the basis of tracking the interactions between the users and the news items. Based on this information, user profiles are constructed that are employed to rank the selected articles with the most relevant items at the top. A general taxonomy of recommendation systems distinguishes collaborative filtering, content-based filtering, and hybrid approaches, each of which uses the information about user interactions with the news articles in a different way (Kunaver and Požrl, 2017). This section briefly describes the most important approaches, with the aim of providing a general understanding of possible approaches to the task of personalized news recommendation.

2.1.1 Collaborative Filtering

In collaborative filtering approaches, the to-be-recommended items are selected by looking at the pool of articles that users with a similar reading history have rated positively (Li and Wang, 2019). A distinction can be made between two types of collaborative filtering, namely user-based and item-based. In user-based filtering, a group of similar users are identified based on, for instance, reading history and topic preferences. The recommendations contain items that similar users have read but that are new to the current user. Item-based filtering identifies similarities between items in terms of content, and recommends the ones with a high similarity to articles that a user has already read.

User-based collaborative filtering is the most widely implemented approach (Burke, 2002). Its main downside is the so-called cold-start problem, where the system cannot recommend new items because they are not yet associated with any users. Similarly, new users cannot receive accurate recommendations as they do not yet have an established preference. Additionally, this approach does not work well for users with unique preferences because they cannot be accurately grouped with other users (Li and Wang, 2019).

2.1.2 Content-based Filtering

Content-based filtering is an extension of item-based collaborative filtering, and establishes recommendations by means of previous interests and clicks of a user. Generally, all users and each item in the recommendation pool are represented by a number of features. For users, the features can include previous likes or dislikes, reading history, and topics of interest based on users' previous reading behaviors. Item features are based on their content. The most relevant items are selected for each user by matching their previous preferences with the available items (Li and Wang, 2019).

Content-based approaches have several advantages compared to collaborative filtering. For instance, it is user-independent, since it does not need evaluation data provided by users, and it is more transparent, as the features are explicitly listed. Moreover, content-based approaches have the ability to recommend items that are new to the collection (i.e., not yet recommended to any user), whereas collaborative filtering cannot do this (Melville et al., 2002). On the other hand, this approach cannot provide recommendations for new users due to lack of information on their preferences. In addition, obtaining informative features to represent the content of the items is expensive in terms of labour (Melville et al., 2002).

2.1.3 Hybrid Approaches

Most state-of-the-art systems use a combination of the two approaches discussed above in order to overcome their drawbacks (Thorat et al., 2015). In most cases, collaborative filtering is combined with other techniques to overcome the cold-start problem, where either new users or new items cannot be taken into account (Burke, 2002). For example, Zheng et al. (2013) use ensemble hierarchical clustering to tackle the problem. Much like user-based collaborative filtering, they group users based on their reading history, with the addition that users may be part of several groups. Additionally, user information (such as reading frequency and demographic information) is used to filter the candidates that are established with the previous step. The combination of enriched group information and user information results in a well-performing system that is not bothered by the cold-start problem, since new users can be initiated in groups based on their profile information.

2.2 Measuring Diversity

The manifesto by Bernstein et al. (2019) calls for rethinking of what diversity means in the context of news recommendations, and the development of evaluation metrics that are in line with this definition. Diversity is a complex normative concept, and its definition can be different depending on the domain. In the context of NRSs, it can be defined as "heterogeneity of media content in terms of one or more specified characteristics" (Bernstein et al., 2019, p. 49). Additionally, the Council of Europe defines diversity as not a goal in itself, but rather a means to promote democratic values (Vrijenhoek et al., 2021). In other words, a diverse recommendation system advocates those values that are deemed important for the enhancement of democracy according to the media organization or news outlet. What these values entail can differ according to different conceptions of democracy. Section 2.2.1 outlines 4 of the most used theories of democracy which take distinctive stances towards the role of media and citizens in a democracy (Helberger, 2019). Section 2.2.2 summarizes the five diveristy metrics that are developed by Vrijenhoek et al. (2021) in response to the manifesto, and relates them to the theories of democracy.

2.2.1 Theories of Democracy

The theories of democracy that are described in this section form a framework in which the goals of a diverse NRS can be analyzed. The framework is developed by Helberger (2019), and contains the Liberal, Participatory, Deliberative, and Critical models of democracy. Since the media play a big role in the every-day life in a democratic society, models of democracy can make the role that media should play in society concrete. As Vrijenhoek et al. (2021) note, one of the democratic models is not inherently better than another. Each model shapes different expectations for the way in which NRSs should inform citizens with important information, as well as the degree to which citizens should actively take responsibility to live up to democratic values. The summaries provided below are adapted from Helberger (2019) andVrijenhoek et al. (2021).

The Liberal Model

The Liberal Model values individual development, freedom and autonomy of citizens, and the right to privacy and freedom of expression. Due to the strong emphasis on personal autonomy, this model favors the view that citizens have the freedom to choose the information they are interested in. Demanding that each citizen spends their time on engaging with the news, politics and public life is too much to ask (Strömbäck, 2005). External influences on the intake of information violate the freedom to choose which information is relevant for them, and should thus be avoided by a NRS. Instead, the role of news outlets should be to inform people of critical problems that require their immediate attention, rather than making sure they have a wide knowledge about developments throughout society. NRSs should thus be interest-driven: citizens should read what they want to know instead of what is decided they need to know.

The Participatory Model

According to the Participatory Model, citizens should actively engage with politics and societal developments to allow society to thrive (Strömbäck, 2005). Community, commitment to citizenship, inclusiveness, equality and tolerance are central, at the cost of self-development, autonomy and ultimate freedom (Held, 2006). All citizens should have a thorough knowledge of the political system, as well as the matters on the political agenda. The media plays a paternalistic role (i.e. the role of an active educator and coach to the public) and should move beyond what people want to read to what they have to read in order to fulfill their civic duty. In this model, diversity calls for the representation of all interests in society, as inclusiveness is crucial. The challenge of a NRS in this context thus becomes to select articles that provide a fair and inclusive portrayal of different ideas and opinions in society. Moreover, the way news is presented should focus on engagement, as it should motivate the public to participate.

The Deliberative Model

Just like the Participatory Model, the Deliberative Model emphasizes the importance of community and active participation of citizens. The difference lies in the Deliberative model's assumption that people's preferences are the result of the active search and comparison of opposing ideas, whereas the Participatory Model assumes that the media has a more paternalistic role of an active educator (Manin, 1987; Ferree et al., 2002). The Deliberative Model thus aims to provide an overview of topics and events from which the users can actively search for their interests as well as opposing views. Values such as open-mindedness, tolerance, equality, and symmetry are central to this model. The role of the media is to both inform the audience on the most important topics and actively confronting them with a diverse array of perspectives, values and opinions that promote discussions and challenge the user's current ideas (Christians et al., 2010). According to the Deliberative Model, a diverse NRS should thus focus on educating the public by representing all relevant perspectives equally, while encouraging the broadening of their horizon. This should lead to polite debate with the goal of coming to an agreement. Moreover, this model has a strong focus on rationality; the news should thus be presented in a neutral tone.

The Critical Model

The Deliberative Model is often criticized for toning down the hidden inequalities in society due to the strong focus on reason and tolerance. By doing so, it can disregard the importance of conflict and disagreement (Karppinen, 2013). The Critical Model aims to move away from the rational presentation of news that is adopted by the other models, and instead opts for the inclusion of emotional and provocative content. The goal is to "escape the standard of civility and the language of the stereotypical middle aged, educated white man" (Young, 2021, p. 176). Moreover, it aspires to amplify the voices and opinions of marginalized group that often go unheard. According to this model, a diverse NRS should thus aim to magnify marginalized voices, and provide articles that move away from the rational and unemotional language that is often used in mainstream media.

2.2.2 Metrics of Diversity

Vrijenhoek et al. (2021) point out that earlier attempts at quantifying diversity often fail to meet a normative definition of diversity that can change depending on the democratic model that is assumed. Therefore, they developed a set of five metrics that reflect the values that play a role in the democratic models: Calibration, Fragmentation, Activation, Representation, and Alternative Voices. The following sections describe the intuition of each metric, as well as the expectations the models of democracy set for the value of the metric. All descriptions below are summaries of the more extensive description from Vrijenhoek et al. (2021). As not all metrics are equally relevant to each theory of democracy, only the models that take a clear stance towards the metric are discussed. A summary of the ideal values of each metric per model of democracy can be found in Table 2.1.

Calibration

The Calibration Score expresses how well a recommendation reflects the preferences of the user. While this is a well-known metric for NRSs, Vrijenhoek et al. (2021) extend it to include not only topicality or genre, but also writing style and complexity. Roughly speaking, it is calculated by establishing the difference between articles that are previously read by the user and what the articles that are present in the generated recommendation set. The extension to writing style and complexity allows recommendations to be tailored to reader's needs more extensively. A set of recommendations that is perfectly calibrated has a score of 0, whereas a large divergence from the reader's preferences is indicated by a score of 1.

In the context of democratic news recommenders, calibration is most relevant for the Liberal and Participatory Models. As the aim of the Liberal Model is to encourage specialization of readers in the topics of their choice, a metric that can detect the articles that best fit their needs is essential. A Calibration Score close to 0 (i.e. a high degree of calibration) is desired in this model. In contrast, the Participatory Model values the common good at the cost of individual preferences. Since the media takes the role of an active educator, the articles that a NRS recommends do not necessarily have to be in line with what the user wants to read, but rather with what is deemed important for them to read. This results in an ideal Calibration Score that is close to 1 in terms of topicality. However, the Calibration Score should be low when regarding complexity, as it is of importance for both models that the articles agree with the level of knowledge the reader has of the topic.

Fragmentation

Fragmentation reflects the degree of overlap between the news story chains that are present in users' recommendation sets. Remember that news story chains are articles that report on the same event or action that took place at a specific place or time. A Fragmentation Score of 0 means that there is a perfect overlap between users: they are all exposed to the same news story chains. This is an indication of a common public sphere, as people are aware of the same events that are happening in society. This allows readers to be exposed to different perspectives, while still being aware of the same issues that play in society. In contrast, a Fragmentation Score of 1 means there is no overlap at all, which indicates the existence of a highly fragmented public sphere where users receive highly specialized recommendations.

A common public sphere is important for the Participatory, Deliberative, so a lower Fragmentation Score is favored. Each of these models require that people are aware of roughly the same topics, as this allows them to form opinions and participate in society. Conversely, the Liberal Model favors the specialization of citizens in topics of their interest, which leads to a higher Fragmentation Score. Since the Fragmentation Score forms a central point in this thesis, the operationalization of this metric is described in more detail in Section 4.4.

Activation

Activation denotes the extent to which an item has the intention to motivate readers to take action on a certain issue. Its most common operationalization is the detection of emotions in a news article, because the intensity of emotions in a text is a good indication of the degree to which a reader can be affected by it. Where an emotional article can motivate readers to take action, a more neutral tone may create more understanding. A score that is close to 1 indicates a high degree of activating content, and a low score means that the content of an article is more neutral.

A NRS that follows the Deliberative Model would avoid recommending too many articles with a high Activation Score, because this model favors polite debates and, eventually, agreement, rather than activism. The Participatory Model prescribes a slightly higher Activation Score, as citizens are deemed to be active citizens that take action when needed. The Critical Model requires a more extreme Activation Score, as it aims to challenge the status quo and values provocative content.

Representation

The Representation metric expresses whether a set of recommended articles is balanced in terms of perspectives and opinions. The score is close to 0 if there is a balance in perspectives, where the definition of a balance is determined by the model of democracy. A score close to 1 indicates that there are large discrepancies in the balance. This measure focuses diversity on *what* is being said, rather than *who* says it, which is captured by the Alternative Voices metric.

For each model, the Representation Score should be low, as it would indicate low divergence from their target distribution. What this distribution should look like differs per model. Since the Participatory Model aspires to reflect the current state of society as closely as possible; the power relations that are present in society should also be present in news recommendations. The more frequent opinions should thus be make up a large share of the recommendations. To the Deliberative Model, balanced representation would entail that each voice is represented equally. The goal of the Critical Model is to shift current power balances by amplifying underrepresented opinions. The ideal distribution of this model would thus be the inverse of the prevalent ideas that the Participatory Model prescribes.

Alternative Voices

This metric evaluates the presence of voices from a minority or marginalized group in the set of recommended articles. More specifically, it measures whether the person or organization that expresses an opinion in an article belongs to a group that is more likely to be underrepresented in mainstream media. There is no universal definition of who is part of a marginalized group, and it can differ per location or context. In practice, this metric is used to give a platform to people with a non-dominant ethnicity, religion, gender identity, sexual orientation or a disability. It is challenging to identify minorities in text, as it requires a lot of contextual information. Pitfalls include unintended stereotyping, misrepresentation, and exclusion by misclassifying target groups. A high score indicates that there is a large presence of minority voices.

The Alternative Voices metric is most important for the Critical Model, as this model by definition focuses on underrepresented voices. The Participatory Model aims to encourage empathy and understanding, and the Deliberative Model favors equal representation of voices, so the Alternative Voices Score should be moderately high.

2.2.3 Summary

In short, the desired value of the diversity metrics changes depending on the model of democracy that is adhered to. Table 2.1 displays an overview the values of each metric that is deemed ideal for each model of democracy. For the metrics that reflect distance of a distribution (namely Calibration and Representation), a “High” target value means that the value of the calculated metric should be close to 0.

	Liberal	**Participatory**	**Deliberative**	**Critical**
Calibration (topic)	High	Low	-	-
Calibration (complexity)	High	High	-	-
Fragmentation	High	Low	Low	-
Activation	-	Medium	Low	High
Representation	-	Reflective	Equal	Inverse
Alternative Voices	-	Medium	Medium	High

Table 2.1: Summary of the desired ranges of the diversity metrics per model of democracy. This table is adapted from Vrijenhoek et al. (2021), p. 181.

2.3 News Story Chains

The Fragmentation metric that is described in Section 2.2.2 forms the main focus of this thesis. It measures the overlap in news story chains that users of a NRS are exposed to. This section provides a more precise definition of the concept of news story chains. It will become clear that there is some debate on the boundaries of a news story chain, leading to differences in terminology. For this reason, a motivation for the use of the term *news story chains* in this thesis is provided. Moreover, this section contains a brief overview of previous work on the task. The aim is not to dive into the technical implementations, but rather to give an intuition of how the task can be approached, and to establish where it can be improved.

2.3.1 Defining News Story Chains

The common level of analysis in the context of NRSs is individual articles. However, news media feeds generally do not consist of individual articles, but rather contain follow-ups on events or incidents. A more useful unit of analysis concerns *news story chains*, which consist of all articles that report on the same action that took place at a specific time and place (Nicholls and Bright, 2019). Apart from follow-up items, news chains can contain more in-depth analyses of the event, opinion pieces, and reports that frame the event in a new or different way. News chains are an especially useful level of analysis in the context of Fragmentation, since this metric measures the degree to which readers are exposed to the same events, not the same individual articles. Potentially, the detection of news story chains can also play a role in extending the Representation metric. If it is possible to properly distinguish between story chains, it can be established whether all relevant voices are represented in the individual chains, and it allows for the identification of chains where this is not the case.

It is not trivial to define the scope of a news chain, since the occurrence of one event might trigger a reaction that can lead to a new event. An event that starts as a news chain might develop into a topic if the relevance and scale accelerates. For example, the first news articles on the war in Ukraine could be regarded to belong to the same event, namely the Russian invasion of Ukraine. However, as the war progresses, more specific news events pop up, such as the siege of Mariupol. The original event has now grown out to be a broader topic that contains various specialized chains. Since story chains are conceptually different from news topics, a clear definition is needed that sets boundaries to the scope of a story chain.

Nicholls and Bright (2019) define news story chains as follows: "events or single issues which receive repeated coverage in the news media through a series of initial articles and follow-up pieces" (p. 43). This definition thus restricts chains to contain multiple articles that report on the same specific event. Moreover, this definition is characterized by a temporal nature: a clear beginning of the chain that triggered the publication of new articles can be identified, resulting in chains with a chronological order. A chain ends when no new publications are added because the event triggers no new developments that receive coverage. In practice, the majority of story chains last a few days (Nicholls and Bright, 2019). Only impactful events, such as the war in Ukraine, are covered for weeks or longer.

Trilling and van Hoof (2020) propose a slightly broader level of analysis instead of news story chains, namely *news events*. They argue that the temporal nature of news story chains is undesired in certain contexts, because it excludes individual articles without follow-ups from the analysis. In the context of, for instance, diversity of news recommendations, single articles can still be meaningful if they report on a specific event. Another nuance in their definition is concerned with the chronology of articles, as information on publication dates is not always relevant. For these reasons, they define news events as "specific events that lead to news coverage" (Trilling and van Hoof, 2020, p. 1321), which can be covered by one or more news articles. In summary, the distinction between news story chains and news events lies mainly in the allowance of single-article events.

In this project, the HeadLine Grouping Dataset (HLGD) (Laban et al., 2021) is used, which contains articles with annotated story chains (see Chapter 3 for a description of the data). For now it is useful to look at the grounds on which the developers of HLGD grouped the news articles. The authors refer to the groups as *timelines*, the definition of which largely correspond to the definition of news story chains as formulated by Nicholls and Bright (2019). Articles are grouped together if they "describe the same event: an action that occurred at a specific time and place" (Laban et al., 2021, p. 3187). Moreover, all articles in the data set are related to a number of other articles, not allowing for single-article chains. Thus, the definition of news story chains by Nicholls and Bright (2019) is adopted in this project.

The above-mentioned definitions do not yet account for the level of granularity of events, i.e. it has no answer to the question of when an event develops into an overarching topic. The scope of events in the HLGD is broader than the datasets used by Nicholls and Bright (2019) and Trilling and van Hoof (2020), as the chains span weeks or even years. Following more conservative definitions, some chains could be subdivided into more specific events. However, dealing with the granularity of events is an open question within this line of research, and should be addressed by developing clear definitions of events and sub-events (Trilling and van Hoof, 2020).

2.3.2 Previous Approaches

Since the tasks of news story chain detection and news event detection are performed in a similar way, this section summarizes previous approaches from both tasks. The task has received relatively little attention. The main reason is that there is no standardized dataset with annotations for news events, which makes it difficult to evaluate the results (Nicholls and Bright, 2019). This section briefly reports on previous work in order to get an intuition of the task, while avoiding technical descriptions for the sake of simplicity.

The most common approach is tackling the task by firstly calculating similarity scores for all pairs of articles in the corpus, and subsequently generating clusters to obtain the news story chains. This approach is adopted by Webber et al. (2010), Boumans et al. (2018), Nicholls and Bright (2019), and Trilling and van Hoof (2020). Each article is paired up with every other article in the dataset. Then, a machine-readable representation of the text is generated. A simple TF-IDF representation is adopted by Webber et al. (2010), Boumans et al. (2018), and Nicholls and Bright (2019). This method is explained in Section 2.4.1, but for now it is enough to know that this method is not quite good at capturing semantic similarity between representations. Trilling and van Hoof (2020) use a more sophisticated representation method, namely word embeddings (see Section 2.4.2).

Once the representations are obtained, similarity scores can be calculated between between the articles. Next, a graph is constructed in which the articles are the nodes, and the similarity scores the edge weights. Edges with a similarity of lower than a certain threshold are generally removed, since it is unlikely that these articles belong to the same chain. As most articles are not similar to each other (Trilling and van Hoof, 2020), the removal of unrelated articles greatly reduces the number of edges. There is no consensus on the optimal value of this threshold. The common approach has been to use graph-based clustering methods to obtain the clusters, because this approach is a suitable choice when dealing with pairwise data (Nicholls and Bright, 2019). Trilling and van Hoof (2020) briefly compared another clustering algorithm (namely hierarchical clustering, which is discussed in Section 2.5.2) to their graph-based algorithm, but found no significant difference in their performance.

The method that combines simple text representations with a graph-based clustering algorithm that is reported here has been found to have have a relatively high precision. However, it tends to be conservative in deciding whether two articles are related, resulting in a lower recall (Nicholls and Bright, 2019; Trilling and van Hoof, 2020). One reason why this type of method often misses articles is the lack of semantic depth that can be encoded (Trilling and van Hoof, 2020). This finding calls for the application of a more sophisticated representation method. One of the contributions of this thesis is providing such an implementation in the form of contextualized sentence embeddings (see Section 2.4.2).

2.4 Text Representation

A common approach for the task of news story chain detection is by means of clustering (Trilling and Schoenbach, 2013). Before the articles can be divided into clusters, their texts should be represented in a machine-readable (i.e. numerical) way. This generally means that the texts should be transformed into vectors that contain numerical values. The quality of the representations greatly affects the performance of the rest of the pipeline (Babić et al., 2020). For example, a clustering method that receives a shallow, surface-form representation of articles might have more difficulty in detecting similarity across texts compared to a representation that contains richer semantic information. Kusner et al. (2015) illustrate this issue by comparing the following two sentences: "Obama speaks to the media in Illinois" and "The President greets the press in Chicago". Although these sentences refer to the same event, this will not be reflected by a similarity measure that only takes surface forms into account, since different words are used to encode the same information. For example, synonyms (different word forms with the same meaning) and hyponyms (words that are specific instances of another) cannot be detected with surface form similarity scores, and will thus receive a low similarity score. For this reason, a way to represent the articles that moves beyond literal forms is required for the current task.

This section describes the text representation methods that are implemented in this project. First, the *Bag of Words* (BoW) method is discussed, which is a simple method to obtain representations based on the words' surface-form. The baseline uses an version of BoW that is enriched with TF-IDF, which is described in the same section. Then, the *embeddings* paradigm is summarized, which is specialized in encoding semantic similarity. Two types of embeddings are discussed, namely word embeddings and sentence embeddings.

2.4.1 Bag of Words

A BoW model is a simple method that represents a document in terms of the presence of its words in a constructed vocabulary. The vocabulary comprises of all words from a corpus (i.e. the collection of all documents), and is thus dependent on the size and variety of the corpus. Each document is assigned a vector with the length of the vocabulary, where a binary indicator marks the presence or absence of the word at that position in the current document. Note that this results in the loss of information on word order. Moreover, identical words with different capitalization will be treated as distinct words.

Since single documents generally contain a small portion of the words that are available in the vocabulary, a BoW model often generates high-dimensional, sparse vectors. The dimensionality is commonly reduced by filtering *stop words* (i.e. the most common words in a language that are equally present in each text, such as 'the', 'to', and 'and'). Stop words have a low information value due to their universally extensive use, regardless of the topic. Various lists of stopwords can be found, generally consisting of function words such as determiners, prepositions, pronouns, auxiliary verbs, conjunctions and conjugations of the verb *to be*.

Adding to the high dimensionality of a BoW model, another downside is its inability to deal with semantic or orthographic similarity. According to this representation, the word *cat* and *rat* are as dissimilar as *cat* and *automobile.* For humans it is clear that *cat* and *rat* have more in common, both semantically (they are both relatively small animals) and orthographically (they differ only in one letter).

TF-IDF

The traditional BoW approach has been enriched in several ways to deal with its shortcomings. For example, Scott and Matwin (1998) extend the representations with WordNet relations (e.g. synonyms, hyponyms and antonyms) to account for semantic relations that holds between words. Another common addition to the BoW representations is TF-IDF (Jones, 1972). It stands for Term Frequency-Inverse Document Frequency, and bases the representation of texts on the words that have a high information value in a document. This follows the intuition that words that occur frequently throughout the corpus are not a good discriminator (Zhang et al., 2011). All words are assigned a value that expresses their degree of informativeness. Words that occur frequently throughout the corpus receive a lower value, while words that are rare in the corpus but relatively frequent in a document receive a higher value.

As its name suggests, the TF-IDF is made up of the multiplication of two scores: the *term frequency* (how often a word occurs in a document), and the *inverse document frequency.* The latter expresses how much information a word carries by counting the frequency of a word throughout the corpus. The count is then reversed to obtain a ranking where the least frequent words ranked high, thus receiving a larger weight, whereas the most frequent words are ranked low. Frequently occurring words such as *the* and *you* are generally present in all texts, and thus are unlikely to provide useful information.

The classical formula for weighing the words is as follows:

$$w_{i,j} = tf_{i,j} \times log\binom{N}{df_i}$$

where w$_{i,j}$ is the weight for term i in document j, N is the number of documents in the collection, tf$_{i,j}$ is the term frequency of term i in document j and df_i is the document frequency of term i in the corpus (Zhang et al., 2011).

A TF-IDF representation encodes documents in a way that allows for comparison between texts in terms of similarity. Texts on related topics are likely to contain an overlap in informative words, which is reflected by a similarity in their vectors. Assigning low weights to uninformative words allows for the identification of these similarities.

2.4.2 Embeddings

With the rise of neural networks, new methods were developed that rethink the traditional way of representing words as vectors. A representation of words could now be learned by considering their context from larger corpora than before. These embedding representations overcome the major drawbacks of a BoW model: high dimensionality and lack of semantic representation. Embeddings are based on the distributional hypothesis, which states that words that occur in the same contexts tend to have similar meanings (Harris, 1954). For example, near-synonyms like *forest* and *woodland* are generally surrounded by similar words (e.g. *trees* or *wild*). The difference in meaning between words can in this way be captured by measuring the difference in their environment.

Embeddings are an improvement of methods like TF-IDF and BoW because they can represent words in a semantically meaningful way. A common test to illustrate this is using arithmetic operations on vectors to finish analogies. Analogies are statements that take the form of "*a* is to *b* as *x* is to *y*". In the analogy "France is to Paris as Italy is to *x*", x can be calculated by subtracting the vectors of France and Italy, and adding the vector of Paris. This will result in the vector of Rome. Embedding models are not trained on this property, they are inherent to the way word vectors are designed (Mikolov et al., 2013c).

However, there are boundaries to the semantic information that embeddings can encode. For instance, they cannot always distinguish between words with opposite meanings such as *bad* and *good.* These words often occur in the same contexts, resulting in close vectors, even though they can completely change the meaning of a sentence. This is mainly problematic for tasks that are concerned with sentiment, which is not the case for the detection of story chains. Another downside is the inability of embeddings to represent words that were not in the corpus on which they were trained.

Embedding representations are traditionally learned at word-level, where a vector representation is constructed for each word. Recently, embeddings at sentence-level are developed, which is especially useful for the task of news chain detection. The following sections provide background information on word and sentence embeddings, with a focus on the models that are used in this project (GloVE and Sentence-BERT respectively).

Word Embeddings

Mikolov et al. (2013b) introduced a method of training a neural network-inspired architecture to learn vector representations of words based on the distribution of their occurrence in a text. Based on the assumption that the meaning of words can be derived from their neighboring words, word embeddings represent words as vectors that map them to a point in a multidimensional semantic space, where numbers in the vector represent coordinates in the vector space. The values of the vectors are commonly learned by neural networks that obtain the representation based on the frequency distributions of words and its neighbors. This results in clusters where words in close proximity of each other occur in similar contexts, and are thus semantically similar. Reversely, a large distance between two words indicates a high dissimilarity (Mikolov et al., 2013c). A variety of algorithms is employed to learn vector representations, among which Skip-Gram, Continuous Bag-of-Words (Mikolov et al., 2013a) and GloVe (Pennington et al., 2014). What follows is a description of the latter, as GloVe embeddings are used in this project.

GloVe (for Global Vectors) is a widely-used, unsupervised algorithm that computes word vectors by means of constructing a co-occurrance matrix of a given corpus, where the rows and columns correspond to the words that occur in the corpus. Each value in the matrix reports the number of times the two words occur together. The count values are then transformed into probabilities that express the likelihood of two words occurring in each others' context. These probabilities are used to train a neural network to learn the weights between all word-pairs that have a occurrence probability of larger than zero. By filtering unlikely pairs, the dimensionality of the embeddings is reduced.

Levy et al. (2015) did an extensive comparative study on popular word embedding models, and found no significant difference in their performance.

A shortcoming of GloVe word embeddings is the fact that they are static, meaning they learn global representations of words where each occurrence of the same word form maps to the same point in the vector space. This makes it impossible for static embeddings to distinguish between the senses of polysemous words. So-called contextualized word embeddings were designed to tackle this problem, where the representation is influenced by the words occurring in its vicinity, thus allowing for different representations of words that can change meaning depending on the context. The following subsection describes an instance of contextualized embeddings, namely those obtained with Sentence-BERT.

Sentence Embeddings

Sentence embeddings are developed to more accurately represent larger chunks of text. As the name implies, sentence embeddings map entire sentences to a vector space, so that similar sentences have similar vectors, and are thus located closely. A simple way to obtain sentence embeddings is by averaging over the sum of the embeddings of the individual words in a sentence. Generally, a step to reduce the dimensionality of the resulting vectors is applied. However, this approach cannot take word order into account, resulting in the same vectors for the sentences “The dog bit Ward” and “Ward bit the dog”.

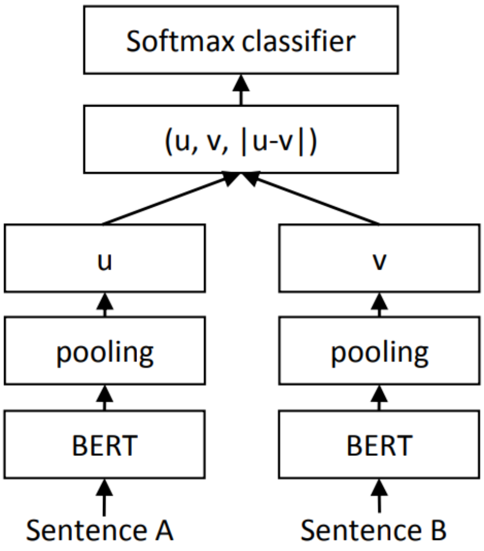

Figure 2.1: The architecture of a Siamese Neural Network

The current state-of-the-art sentence embeddings are developed by Reimers and Gurevych (2019). They use a transformer-based language model to learn the vector representations. Among the first and most popular transformer models is BERT (Devlin et al., 2019). In short, a transformer is a deep learning model that is specialized in sequential data. Its defining feature is the self-attention mechanism it employs to call upon important information from preceding states, such that information from the entire input sequence is available throughout the pipeline. Reimers and Gurevych (2019) adapted BERT to contain a Siamese neural network, which is an architecture that contains two or more identical networks that have the same configuration but different inputs (see Figure 2.1 for a visual illustration of the architecture). The two BERT networks (one for each input) have tied weights, meaning that they run through the same configuration with the same weights. The aim of a Siamese network is to calculate the similarity between the inputs. By providing sentences as input for the Siamese neural network, BERT will generate embeddings for each of them downstream. The resulting sentence embeddings represent the interaction between words in a semantically plausible way, and can take the word order into account.

2.5 Clustering Methods

This section summarizes three classes of clustering methods: graph-based, hierarchical, and density-based clustering. A graph-based algorithm has been applied to the detection of story chains in precious work (see Helberger (2019), Trilling and van Hoof (2020), and Vrijenhoek et al. (2021)). This approach functions as the baseline, as it is also used to detect story chains in Vrijenhoek et al. (2021). The hierarchical and density-based approaches are widely used clustering methods that have a property that make them suitable for the task, namely the possibility of leaving the number of clusters unspecified. Since news feeds are continuously updated as new events happen, there is no finite number of clusters: a newly added news article might be the start of a new cluster. Both agglomerative hierarchical clustering and density-based approaches satisfy this characteristic, as there is no requirement for a pre-defined number of clusters (Bouguettaya et al., 2015).

2.5.1 Graph-based clustering

Graph-based clustering methods aim to construct a network-like representation of the data points, where each datapoint is a node, and the edges between nodes contain a similarity score. This task is often referred to as *community detection* (Lancichinetti and Fortunato, 2009). The graph is partitioned into subgraphs by menas of an algorithm where edges *within* a subgraph should have high weights (i.e., high similarity), whereas edges *between* subgraphs should have low weights (i.e., low similarity) (Chen and Ji, 2010). The resulting partition is hierarchical, as it contains multi-level clusters, where subgraphs are part of a larger graph.

A variety of algorithms can be applied to obtain the partitioned graph, see Chen and Ji (2010) for an overview. Vrijenhoek et al. (2021) employ the Louvain algorithm (Blondel et al., 2008). This method, combined with the TF-IDF representations, forms the baseline of this thesis, as one of the aims is to improve the technical implementation of the Fragmentation metric.

The Louvain algorithm is a greedy optimization method that aims to optimize the *modularity* of the partition. The modularity measures the density of the links *within* communities and compares them to the links *between* communities. The optimization is a top-down, iterative process that consists of two phases. In the first phase, each node in the network is assigned to a different community (i.e. cluster). Then, each node is merged with the neighbor that leads to the maximal gain in modularity. This process is repeated until there is no room for improvement. In the second phase, a new network is built, where the nodes are the communities that are found in phase one.

The weights of the new links are calculated by summing the weights between the nodes it contains. Now, the process can be repeated, until no improvements in modularity can be found.

This algorithm is suitable for the task of news chain detection because it does not require a pre-specified number of resulting clusters. Moreover, the hierarchical output is informative in this context, since it may be able to capture topics that contain multiple story chains. The remaining sections describe other clustering methods that satisfy the characteristic of not having to specify the number of predicted clusters.

2.5.2 Hierarchical Clustering

Hierarchical clustering is a widely used clustering algorithm. A distinction can be made between agglomerative and divisive hierarchical clustering. Agglomerative clustering is a bottom-up approach, where each data point initially represents a cluster. Divisive clustering creates the hierarchy in a top-down fashion, with one large cluster that repeatedly gets split into smaller ones (Murtagh and Contreras, 2012). Due to the ease of implementation, agglomerative hierarchical clustering was implemented in this project. The remainder of this section describes this type of clustering in more detail.

Agglomerative hierarchical clustering merges data points based on their similarity. Firstly, each data point is assigned to a new cluster. Then, pairs of clusters are recursively merged based on similarity. The similarity is generally calculated by means of a similarity matrix that is calculated with similarity scores such as cosine. This results in a dendogram, which is a hierarchical, tree-based representation of a complete cluster that contains all data points organized in sub-clusters. The leaves correspond to individual data points, in this case articles, and the nodes represent the clusters. Two groups that are merged receive a new internal node. An advantage of this method is that it allows for exploration of the clusters on different levels of granularity (Berkhin, 2006). This can be especially useful in the context of news story chains, since the question of when a story chain becomes a broader topic containing multiple story chains is an open question.

The decision of which clusters will be merged into subclusters depends on two parameters, namely the *distance threshold* and the *linkage criterion.* The distance threshold specifies the value above which two objects will not be merged. A large distance between two objects signifies a high dissimilarity, whereas objects that are more similar result in a smaller distance. The way this distance is calculated is determined by the linkage criterion, which calculates the distance between all pairs of points where one point is in the first cluster, and the other in the second. This can be done in a variety of ways, but the most common linkage criteria are Ward's linkage, complete linkage, average linkage, and single linkage (Berkhin, 2006).

The *single linkage* (or nearest neighbor) criterion defines the distance between two clusters as the distance between the closest possible points. It has as an advantage that it is efficient to compute, but its pitfall is the so-called phenomenon of *chaining.* This is the process in which items are incorrectly judged as similar through transitivity: if item A is similar to B, and B is similar to C, A is not necessarily like C. However, the single linkage approach does employ such transitivity chains, leading to inaccurate clusters, especially at the higher levels of the dendogram (Aggarwal and Zhai, 2012).

Average linkage clustering takes the average distance between all points in two clusters, and is the least affected by outliers. In *complete linkage* clustering, the similarity between two objects is defined by the distance between the furthest points of the two clusters (Aggarwal and Zhai, 2012). Although these take longer to compute, they are more robust than single-linkage clustering because they do not fall prey to chaining.

Ward's linkage takes a slightly different approach. The distance between two clusters is expressed as the increase in the *error sum of squares* (ESS) after merging two clusters into one. Ward's linkage minimizes the ESS in a step-wise manner by merging the two clusters that have the smallest cost to merge (de Amorim, 2015).

2.5.3 Density-based Clustering

One disadvantage of hierarchical clustering is that is assumes that each data point is relevant: it is unable to identify singleton clusters. However, in the task of story chain detection, single-event chains are a possibility (although not in the current data set). For the task in general, it is insightful to explore the performance of a density-based clustering approach, which is able to construct singleton clusters. Moreover, density-based approaches can deal with clusters of different shapes and sizes, whereas hierarchical clustering methods, especially with Ward's linkage, generally proposes clusters of equal size. In other words, hierarchical clustering tends lump clusters together if it results in a more balanced distribution, whereas density-based can produce clusters varying in size, and is more likely to identify singleton clusters. This section describes the most popular density-based clustering method, DBScan (Ester et al., 1996). Just like hierarchical clustering, it does not require a pre-specified number of clusters.

The workings of density-based approaches are intuitive: areas with a high density of data points are merged into clusters. The algorithm first identifies *core points* which are located in the neighborhood of a minimal number of data points n that is to be specified by the user. The maximal distance between two points to be considered neighbors is also specified by the user. All core points that are linked by proximity will be merged into a cluster. Once all core points are identified, *border points* are identified. They do not fulfill the requirement of having at least n points in their proximity, but they should be close to at least one core point. Once all of them are identified, they merge with the cluster that the closest core point is a part of. Lastly, *noise points* are identified, which are not in the proximity of any core point. It can be close to a border point, but will not be able to join the cluster that this border point belongs to, because merging is only possible when at least one core point is involved. In other words, border points can only join a cluster, but they cannot extend them.

2.6 Cluster Validation

The goal of a clustering algorithm is to find partitions in the input data resulting in clusters, where the objects within a cluster are similar, while objects in different clusters are dissimilar (Arbelaitz et al., 2013). Assessing the quality of the partitioning is an important step in the clustering process because the outcome of a clustering algorithm relies heavily on parameter settings (e.g. the k parameter for specifying the number of desired clusters). The process of estimating how well a proposed partition fits the underlying data is known as *cluster validation.* Defining the quality of the clustering output can be subjective, because what it means to be of high quality depends on the goal of the project and the underlying data (He et al., 2004). For the purposes of story chains in NRSs, an individual cluster is of high quality if it mostly contains articles that belong to the same story chain. Overall, the clustering output is of high quality if most members of a story chain are assigned to a single cluster, instead of over multiple clusters.

A common distinction of validation metrics is between *internal* and *external* validation. External metrics compare the proposed partition with the correct partition, and thus can only be performed when the gold standard labels are known. Internal metrics base the quality only on the resulting partitions. In this project, external metrics can be used because the story chains that an article belongs to are known. It would nevertheless be interesting to see how the external evaluation relates to internal metrics, because it allows for comparison to previous projects that only use internal metrics. The remainder of this section describes different internal and external metrics, and outlines their strengths and weaknesses.

2.6.1 Internal Cluster Validation

Arbelaitz et al. (2013) performed a large-scale comparative study on 30 internal cluster validity measures. This type of internal metric estimates the quality of a partition by measuring the *compactness* and *separation* of the clusters. Compactness refers to the degree to which objects in a cluster are related, and is often measured in terms of variance. A low variance indicates a high compactness, and is generally desirable (Liu et al., 2010). However, this metric incorrectly values clusters that contains a single object with a compactness of 1, as it contains low variance, even though this partition does not reflect the underlying data. For this reason, compactness measures are usually complemented with a measure for separation that analyzes how well-separated partitions are. Again, this is often measured in terms of variance, where a high variance between clusters indicates a high degree of separation. Most of the measures analyzed in Arbelaitz et al. (2013) use a combination of compactness and separation to assess the quality of a partition.

They found no sufficient evidence supporting that some metrics capture the validity of the clusters significantly more accurate than others across datasets and configurations. However, the Silhouette (Rousseeuw, 1987), Davies–Bouldin (Davies and Bouldin, 1979) and Calinski–Harabasz (Caliński and Harabasz, 1974) measures have been found to be the most reliable metrics, as their performance was relatively stable across configurations, and are thus preferred (Arbelaitz et al., 2013). Since the performance of internal validity measures varies highly across datasets and configurations, it is recommended to use several measures to increase the robustness and stability of the evaluation. We include the Silhouette Score, because it can provide an easily interpretable visualization of the clusters. While the Davies-Bouldin and Calinski-Harabasz scores express similar notions, the former is the least computationally expensive to calculate and will thus be included in the analysis. Below follows a description of each metric.

Silhouette Score

The Silhouette Score (Rousseeuw, 1987) measures the validity of clusters by calculating how similar each object is to other data points in the cluster it is assigned to compared to closest adjacent clusters. A coefficient is calculated for each data point. The average of all coefficients results in the Silhouette Score for the clustering outcome. The score can range between -1 and 1, where a high value indicates that the object is similar to the cluster it is assigned to, and dissimilar to other clusters. A value around 0 indicates overlapping clusters, and a negative value indicates incorrect clustering.

The Silhouette coefficient is calculated with the following formula, where a(i) is the mean distance between a point and all other points in the same cluster, and b(i) the mean distance between a point and all points in the nearest cluster.

$$s(i) = \frac{b(i) - a(i)}{max(b(i) - a(i)}$$

An advantage of this score is that the plotting of all coefficients results in an interpretable visual representation of the clusters. This also allows for the identification of outliers. A drawback is that the score tends to be higher when dealing with density-based clusters.

Davies-Bouldin Index

Similar to the Silhouette Score, the Davies-Bouldin Index (Davies and Bouldin, 1979) compares the average similarity of each cluster and its closest neighbor. Here, the similarity is the ratio of distances within a cluster to distances between a cluster. Clusters that are further apart will result in a score that is closer to 0, which indicates well-separated clusters. This metric is less computationally demanding than the Silhouette Score, but it has the same drawback of generally providing better scores for density-based clustering techniques.

2.6.2 External Cluster Validation

External cluster validation measures compare a clustering outcome to its gold labels. Rosenberg and Hirschberg (2007) propose an external metric that is based on the notions of *homogeneity* and *completeness*. A proposed partition is perfectly homogeneous if all clusters contain only members of a single class. Completeness is satisfied if all members of a given class are assigned to the same cluster. The harmonic mean of the homogeneity and completeness scores result in the *V-measure*. This measure captures both the elements that are important in for the evaluation of chain detection, namely how clean each cluster is, and how the members of a story chain are distributed over all proposed clusters.

The V-measure has advantages over other external cluster validation methods such as the F-measure, which is the harmonic mean of precision and recall scores. In the context of story chain detection, the precision of a partition would express the proportion of articles that are correctly assigned to the same cluster. Recall would indicate the proportion of all articles that were assigned to the correct cluster. Although this metric measures homogeneity (expressed by precision) and completeness (expressed by recall), it suffers from a critical problem that is referred to as the “problem of matching” (Rosenberg and Hirschberg, 2007, p. 410). The problem arises because only the majority class in a cluster is evaluated with the F-measure. In other words, two partitions with the same number of correct class members, but different incorrect class members, can result in the same score. The F-measure thus fails to provide insight into the mistakes that a partition contains. Another advantage is that the V-measure can be calculated independently from the absolute values of the true labels, whereas the F-measure requires a one-on-one mapping of predicted cluster labels and true class labels. This mapping requires a post-processing step where the predicted labels are transformed to match the gold labels, which is not always possible (e.g. when a proposed partition contains more clusters than the true clusters).

Two other metrics that are often used in the context of external cluster validation are *Purity* and *Entropy*. Purity is similar to homogeneity in the sense that it expresses the percentage of a cluster that is occupied by the majority class. The Purity of a proposed partition is the weighted sum of the purity of individual clusters, where a high value is desirable (Abualigah et al., 2018). Entropy expresses how members of the same class are represented by the various proposed clusters (He et al., 2004). A low entropy indicates that classes are homogeneously distributed, and is thus an indication of a good clustering outcome. However, Purity and Entropy lack a way of accounting for completeness, as it is not measured whether a single cluster contains all members of a certain class. A clustering outcome that assigns each data point to its own cluster will score high on Purity and low on Entropy, but this partition is rarely the desirable outcome (Rosenberg and Hirschberg, 2007).

Chapter 3

Data description

Finding annotated datasets for the task of story chain detection is relatively difficult, since the annotations are expensive to generate (Nicholls and Bright, 2019). For each pair of two articles, annotators should decide whether they belong to the same story chain or not. Both Trilling and van Hoof (2020) and Nicholls and Bright (2019) use a data set of which only a small portion is annotated. To guarantee the evaluation of the pipeline proposed in this project, a fully annotated data set is required. For this reason, the HeadLine Grouping Dataset (HLGD) (Laban et al., 2021) is used, which contains headlines of English news articles on a variety of topics. Each article is paired up with every other article in the dataset, forming pairs of which the relation is annotated. The dataset is described in more detail below.

3.1 HeadLine Grouping Dataset

The HLGD (Laban et al., 2021) contains 1679 articles that are divided into 10 story chains ranging in size from 80 to 274 news articles, and span 18 days to 10 years. Note that the time spans of the chains are considerably larger than the conventional definition by Nicholls and Bright (2019). The reason for this is that the topics that articles report on are more impactful or controversial than the topics in the datasets of Nicholls and Bright (2019) and Trilling and van Hoof (2020). Figure 3.1 displays the news story chains that are present in the data set, along with the number of articles that belong to each chain.

The articles are time-stamped, and originate from 34 different news sources. The annotations of the story chains were provided by five independent annotators that achieved an average inter-annotator agreement score of 0.814. The majority of article pairs are negative, because two random articles are unlikely to be reporting on the same story chain. To fight this imbalance, the authors down-sampled negative pairs, resulting in a ratio of 1 positive pair per 5 negative pairs, which is in line with similar datasets (Laban et al., 2021).

3.1.1 Article Extraction and Preprocessing

The original dataset only contains the titles of the articles and the corresponding URL, because the authors performed the task of chain detection with solely the headlines as input. Since the link was provided, the full text could be scraped with the Trafilatura package (Barbaresi, 2021). In the process of extraction, 285 articles had to be excluded

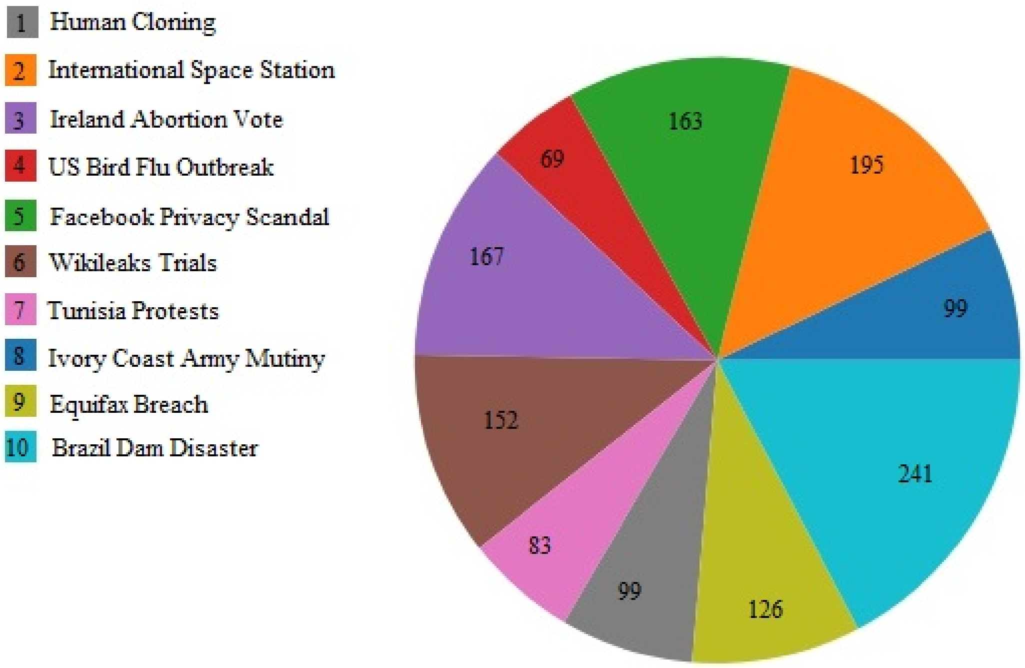

Figure 3.1: Topics in the HeadLine Grouping Dataset. Each news story chain is numbered, and the total count per chain is displayed.

due to one of the following reasons. Some links could not be accessed due to paywalls, or because the articles were removed. For other links, the CAPTCHA message asking the visitor to indicate they are not a robot was scraped instead of the article text. Two articles with a length of 100.000 each, which appeared to be in-depth interviews, were removed because they would add noise. In a rare case, the HLTM code could not be parsed correctly. A handful of articles consisted of recommendations from the editor, containing multiple headlines about various topics, which were removed because they do not report on a single event. A few dozen articles contained a list of recommended titles at the end of the article. These recommendations were removed from the article text to avoid adding noise to the data, but the rest of the text was left intact.

The cleaned data set consists of 1394 articles, which have a mean length of 3396 characters and a standard deviation of 2247. The shortest article contains 252 characters, and the longest 24988. A total of 145 articles are shorter than 1000, whereas 9 articles are longer than 10000. Table 3.1 displays the number of articles that are present in each story chain, as well as their mean number of characters and tokens per news article.

The corpus is split into a development and evaluation set. The development set contains 363 articles from chains 1, 2 and 4 (see Table 3.1). The size of the clusters was taken into account when constructing the development set, such that it contains clusters of varying sizes. The evaluation set comprises 1031 articles from the remaining news story chains.

#	**Topic**	**Size**	$\overline{x}$ **Characters**	$\overline{x}$ **Tokens**
1	Human Cloning	108	4354	805
2	International Space Station	215	3141	597
3	Ireland Abortion Vote	170	4134	787
4	US Bird Flu Outbreak	75	2266	422
5	Facebook Privacy Scandal	172	4098	763
6	Wikileaks Trials	153	7398	1390
7	Tunisia Protests	86	3201	593
8	Ivory Coast Army Mutiny	104	2231	417
9	Equifax Breach	156	4041	744
10	Brazil Dam Disaster	247	2970	564

Table 3.1: Topics of the news story chains in HLGD, the number of articles in each chain, and the mean number of characters and tokens of the articles per chain.

Chapter 4

Methods

This thesis aims to answer two questions, namely (1) how various representation methods and clustering algorithms compare on the task of news story chain detection, and (2) how the resulting Fragmentation Score is affected by variations in the chain detection system. To answer these questions, a pipeline is developed with four components that are vizualised in Figure 4.1. The first two components aim to answer the first research question, and consists of two steps: (1) representing the news articles with various methods, and (2) clustering articles into news story chains. The remaining two components answer the second research question, and consist of: (3) generating sets of news recommendations for simulated users based on three scenarios; and (4) calculating the Fragmentation Score over the resulting combinations. The following sections describe each step and report on intermediate results (e.g. hyperparameter tuning) where necessary. All code can be found at
github.com/aapolimeno/ClusteringFragmentation

The four components of the experimental setup are as follows:

1. *Article representation* with three methods: Bag of Words (referred to as BoW), word embeddings with GloVe (referred to as GloVe), and sentence embeddings with Sentence-BERT (referred to as SBERT).

2. *Clustering* to obtain story chains with three approaches: graph-based clustering, agglomerative hierarchical clustering and density-based clustering with DBScan.

3. *Generating news recommendations* for three scenarios: low Fragmentation, high Fragmentation and balanced Fragmentation

4. *Calculating Fragmentation Scores* for all outcomes of the previous steps in the pipeline.

4.1 Article Representations

Before the articles can be grouped into clusters, they should be represented as machine-readable vectors. This section describes the representation methods that are used in this thesis, namely Bag of Words (Section 4.1.1), GloVe word embeddings (Section 4.1.2), and Sentence-BERT sentence embeddings (Section 2.4.2). The methods vary in their ability to capture semantic similarity between texts, where BoW representations are poorest in terms of informativeness. Word embeddings are richer than BoW, but not richer than sentence embeddings.

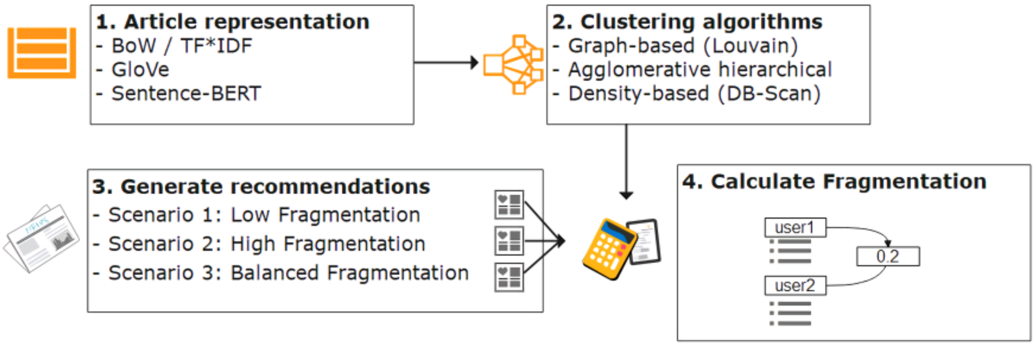
1. Article representation
- BoW / TF*IDF
- GloVe
- Sentence-BERT
2. Clustering algorithms
- Graph-based (Louvain)
- Agglomerative hierarchical
- Density-based (DB-Scan)
3. Generate recommendations
- Scenario 1: Low Fragmentation
- Scenario 2: High Fragmentation
- Scenario 3: Balanced Fragmentation
4. Calculate Fragmentation
user1
0.2
user2

Figure 4.1: A visualization of the experimental setup. The resulting story chain predictions from combinations of the methods in step 1 and 2 are assigned to the news article recommendation sets generated in step 3. In step 4, the Fragmentation Score is calculated over each combination of methods.

4.1.1 Bag of Words

The first approach to represent the articles is a Bag of Words (BoW) model (see Section 2.4.1). The vocabulary is constructed by extracting all words that occur across all articles. Then, each article is represented as a sequence of binary indicators for each word in the vocabulary. This results in a sparse matrix, where the dimensionality is equal to the number of words in the vocabulary. Stop words were excluded in order to reduce the dimensionality. Additionally, all texts are converted to lowercase, because a BoW model would treat the same words with different capitalization as distinct words. Lastly, all punctuation was removed, resulting in a final vocabulary of 19046 words. Since a BoW model is not able to capture semantic similarity, it is expected that the results for this representation method will be low compared to the embedding representations.

4.1.2 Word Embeddings with GloVe

This method represents each word as a vector that maps the word to a point in a multidimensional space. Similar words are located in close proximity of each other, while dissimilar words have more space between them. See Section 2.4.2 for a more elaborate description. The pre-trained GloVe embeddings (Pennington et al., 2014) are used to obtain word embeddings for each word in the corpus. The vectors for the articles are obtained by averaging over the embeddings of each word in the article. GloVe embeddings have a fixed dimensionality of 300, and are thus much smaller and less sparse than the BoW representations.

An important characteristic of GloVe embeddings is that they are static: they learn global representations for each word that does not change depending on the context. This method is thus unable to distinguish between the senses of a polysemous word. Moreover, information on the word order is lost due to the averaging over word embeddings to obtain the article representations.

4.1.3 Sentence Embeddings with Sentence-BERT

Sentence embeddings overcome some of the shortcomings of word embeddings. For instance, by mapping entire sentences to points in a vector space, word order can be retained. In this thesis, the pre-trained sentence embeddings are obtained from Reimers and Gurevych (2019), which have a dimensionality of 384. Documents are treated as single sentences, and thus receive one representation each.

This method is expected to outperform the other representations, because of its ability to capture semantic information beyond word-level, which allows for more complexity. Moreover, the sentence embeddings are contextualized, meaning that the vector of a word is influenced by the words in its vicinity. This results in different representations depending on the context a word occurs in, allowing for a more accurate semantic representation.

4.2 Clustering News Story Chains

This section summarizes the workings of the three clustering methods, and reports the outcome of hyperparameter tuning on the development set. The performance was evaluated with the V-measure (Rosenberg and Hirschberg, 2007). It expresses the harmonic mean of the homogeneity and completeness scores. Homogeneity is high when each cluster contains members of a single class, whereas completeness is high when all members of a given class are assigned to the same clusters (see Section 2.6). In most cases, a range of parameter values produced the same V-measure, and were thus all applied to the the remainder of the data. The settings with the highest performance are reported in Section 5.1.

4.2.1 Baseline

The baseline comprises the partitions as predicted by the Louvain Community Detection Algorithm (Blondel et al., 2008), the graph-based clustering method employed by Vrijenhoek et al. (2021). Firstly, article pairs are constructed, containing combinations of an article with each other article in the data set. Secondly, a TF-IDF representation of the articles is created. Thirdly, the cosine similarity between the representations of the pairs is calculated. If the cosine similarity score is below 0.5, is was assumed unlikely that the pair belongs to the same chain, and is thus dropped from the rest of the clustering process. Lastly, the Louvain Community Detection Algorithm identifies the optimal partitions by maximizing the density within clusters and minimizing the density between clusters, as described in Section 2.5.1.

The resulting partition cannot produce singleton clusters, so articles that were not found to be related to any other article do not receive a class label, but are left out of the partition. Since the number of articles in the gold clusters and the predicted clusters should be the same, the 9 articles that were left out by this method are assigned to its own cluster.

4.2.2 Agglomerative Hierarchical Clustering

As outlined in Section 2.5.2, agglomerative hierarchical clustering is a suitable approach when the number of clusters is unknown. Another advantage is that the clusters are ordered in a hierarchy, where each cluster subsumes multiple smaller clusters (Murtagh and Contreras, 2012). Conceptually, this is in line with the structure of news story chains, where an overarching chain might consist of articles that report on sub-events that link back to the chain event. Depending on the working definition of a chain, these sub-events might form a new chain. The fact that hierarchical clustering can capture the different scopes of story chains makes it a natural approach to the task.

The clustering outcome largely depends on the settings of two hyperparameters: the *distance threshold* and the *linkage criterion.* The distance threshold represents the distance between clusters above which they will not be merged. The linkage criterion specifies how the distance is calculated (e.g. between the most central point in each cluster, or the furthest points). The following options are available: Ward's linkage, average linkage, complete linkage, and single linkage.

The optimal settings of both hyperparameters were determined by testing the performance of a range of values on the development set. The distance threshold ranged from 1 to 150, with steps of 1. The experiment was repeated for each linkage criterion mentioned above, resulting in 1788 combinations that were evaluated. The best results are displayed in Table 4.1. In all cases, Ward's linkage overwhelmingly outperformed the other linkage criteria, as it makes up the top 45 best performing setups.

Model	**Distance**	**Linkage**	**V-measure**
SBERT	5 - 9	Ward	0.962
Word	4 - 6	Ward	0.885
BoW	124 - 148	Ward	0.882

Table 4.1: The best-performing hyperparameter settings of the agglomerative hierarchical clustering approach on the development set.

4.2.3 DB-Scan

Just like hierarchical clustering, the density-based approach DB-Scan (Ester et al., 1996) does not require an indication of the desired number of clusters. Moreover, it can produce clusters in varying shapes and sizes, whereas hierarchical clustering tends to propose clusters of equal sizes and thus lumps clusters together if it results in a more even distribution. The intuition of DB-Scan is that it detects areas with a high density of data points and merges them into clusters. First, core points are identified, which are points that are in close vicinity of at least n other points, where n is specified by the user. Then, border points are distinguished, which are close to a core point, but not necessarily to other points. These points can join a cluster, but they cannot extend them as core points do. Lastly, noise points are identified, which are not in the proximity of a core point, and will not be merged to any cluster.

The algorithm has two hyperparameters that determine the clustering outcome. The most influential parameter is epsilon (*eps*), which expresses the maximum distance between to points to be regarded as being in each other's proximity. Additionally, the minimum numbers of data points in a neighborhood that is necessary for a point to be considered a core point has to be specified, and is denoted by the parameter *min_samples.*

Again, an experiment was performed in which combinations of the two hyperparameters were tested and evaluated on the development data. The best-performing settings in terms of V-measure are displayed in Table 4.2. Values for each parameter ranged between 1 and 150, with steps of 1, resulting in 29403 combinations. The V-measure again stayed stable over a range of values, especially concerning the minimum number of samples.

Model	**Eps**	**Min_samples**	**V-measure**
SBERT	1	32 - 66	0.883
Word	1	1 - 99	0.026
BoW	9 - 12	2	0.320

Table 4.2: The best-performing hyperparameter settings of the DB-Scan approach on the development set.

4.3 Generating News Recommendations

The next step in the pipeline is concerned with generating sets of news recommendations over which the Fragmentation Score can be calculated. As outlined in Section 2.1, NRSs generally make use of user information such as previous reading behaviour to predict which articles they are most likely to read. However, there is no dataset in which both user information and story chains are annotated. As both are necessary for the calculation of Fragmentation, the recommendations have to be simulated.

We considered multiple options for the simulation of user data, among which the matching of an existing data set of news articles that has annotations for user information with HLGD articles. We designed a setup in which articles from the HLGD and a corpus with user interactions (e.g. clicks, likes, and comments) are matched in terms of similarity. The user information was transferred to the matching HLGD articles, such that an existing NRS could generate recommendation sets. However, this approach was abandoned because the articles in the two corpora were too different and could not straightforwardly be matched. Moreover, this would add two steps to the pipeline (enriching the data set with user information, and generating news recommendations) that cannot be accurately evaluated. Its effect on the final Fragmentation Score can thus not be straightforwardly measured, making this approach unsuitable.

A more simple approach concerns the construction of recommendation sets based on scenarios in which an estimation of the resulting Fragmentation Score based on the gold labels can be made. For example, in one scenario, all users are interested in all story chains, and thus read at least one article from each chain. For each reader, the recommendation set contains articles from each story chain, although the specific articles may differ, resulting in a low Fragmentation Score. We thus create three scenarios that lead to different Fragmentation Scores, which allows for the verification of the scores that are generated by the clustering methods. Adding the predicted cluster labels will result in different Fragmentation Scores per setup, which can then be compared to the gold Fragmentation Score to gain insight into the effect of previous pipeline steps on the score.

A total of 1000 users are simulated per scenario, as this should be enough to generalize. In each scenario, the recommendation sets contain 7 articles that are randomly sampled based on different expectations. To account for variation resulting from the randomly selected articles, the process of generating recommendation sets is repeated 10 times. The random samples are based on the true news story chains, but the labels that the experimental setups assign to the selected articles may vary. To account for differences in the resulting Fragmentation Score, the Fragmentation Score is calculated over the 10 recommendation sets with different random samples.

Each scenario is described below, and summarized in Table 4.3. The first two scenarios are ends of a spectrum where all readers either tend to read one article from each story chain, or prefer to specialize in one chain. The last scenario provides a more realistic scenario, where different user profiles are constructed based on reading preferences, resulting in a more balanced Fragmentation Score.

Scenario	**Chains per user**	**Fragmentation**
Scenario 1	7	Low
Scenario 2	1	High
Scenario 3, profile 1 (70%)	5	Balanced
Scenario 3, profile 2 (15%)	2	Balanced
Scenario 3, profile 3 (15%)	7	Balanced

Table 4.3: Overview of the number of chains that are present in the recommendation sets per scenario. In each scenario, there are 1000 users who receive a recommendation set containing 7 articles. Scenario 3 is build with 3 distinct user profiles that differ in the amount of story chains users are exposed to.

4.3.1 Scenario 1: Low Fragmentation

In this scenario, users have a broad interest in all topics present in the dataset. Each recommendation set contains one article from each gold story chain, where the selection of the specific articles is randomized. This results in recommendation sets of 7 articles for each of the 1000 users. The corresponding story chain labels as predicted by each method are obtained. When the gold labels form the input of the Fragmentation function, the resulting score will be 0, as this scenario captures a perfectly heterogeneous distribution of story chains. The Fragmentation Score for the recommendation sets paired with the predicted labels by the experimental clustering methods will differ depending on the method's ability to correctly identify the story chains.

4.3.2 Scenario 2: High Fragmentation

The second scenario simulates a situation in which each user reads articles from a single chain (although the chains differ among users), resulting in a small overlap in the recommended stories. The users are evenly distributed over the 7 chains, where each group reads 7 articles from the chain they are assigned to. In this way, users only have a perfect overlap with users within their group, but no overlap at all with users from other groups. This results in a aggregated Fragmentation Score of 0.85 for the gold labels. It is not possible to construct a Fragmentation Score of 1 for the gold chains, because this would require no overlap at all between the chains that are recommended to the users. Since there are 1000 simulated users, and only 7 story chains in the evaluation set, there will always be some degree of overlap between users.

4.3.3 Scenario 3: Balanced Fragmentation

The third scenario aims to simulate a more realistic news diet. It is difficult to predict the reading behaviour of users in the context of story chains, as there is no literature on this topic. However, there is literature on the fragmentation of news outlets. According to Trilling and Schoenbach (2013), people tend to “choose a comprehensive news diet, including a bit of everything from a broad range of sources” (Trilling and Schoenbach, 2013, p. 947). A simulation of a realistic scenario thus includes a large portion of readers that have a broad interest and read articles from multiple chains. A smaller portion chooses articles from either a small or large number of news chains.

Based on the observation that most people tend to select a variety of articles, three user profiles were constructed. *Profile 1* comprises 70% of the users, who read at least one article from 5 story chains. The 5 story chains are randomly selected per user. To obtain the total number of 7 recommendations per user, a sample of 2 random articles from 2 of the 5 selected chains is added. *Profile 2* consists of 15% of the readers, who have a more specialized preference and read 7 articles from 2 randomly sampled story chains, where 4 articles come from one chain, and 3 of the other. In *profile 3*, comprising the remaining 15%, the readers have an exceptionally broad interest and read one article from each story chain, like in scenario 1. The resulting Fragmentation Score according to the gold labels is 0.58.

4.4 Calculating Fragmentation

The next step consists of calculating the Fragmentation Score for each scenario and clustering setup. The scenarios are made by grouping articles into recommendation sets based on the gold story chain labels. The labels as predicted by each clustering setup described in Section 4.2 are subsequently added to each scenario, which will result in different Fragmentation Scores depending on the performance of the clustering systems. The remainder of this section describes how the Fragmentation Score is calculated, and provides more details on the procedure.

The Fragmentation metric as developed by Vrijenhoek et al. (2021) is implemented. It takes two lists with ranked recommendations as input, where the story chain of each recommendation is specified. The Fragmentation Score is defined as “the aggregate average distance between all sets of recommendations between all users” (Vrijenhoek et al., 2021, p. 177). It is based on the Rank Biased Overlap (Webber et al., 2010):

$$RBO(Q_1, Q_2, s) = (1 - s) \sum_{d=1}^{\infty} s^{d-1} \cdot A_d$$

where Q1 and Q2 represent the ordered recommendation lists, s a parameter that ensures that the recommendation at place 1 is weighted more heavily than lower-placed recommendations. The average overlap A_d is calculated by iterating over all ranks d. This results in a score between 0 and 1, where 1 indicates a perfect overlap. Vrijenhoek et al. (2021) inversed the score to make it more compatible with the other metrics they proposed, so that the Fragmentation Score is calculated by taking 1 minus the Rank-Biased Overlap. With this formulation, a Fragmentation Score of 0 indicates a perfect overlap between readers, whereas a Fragmentation Score of 1 indicates completely disjoint recommendations. The aggregated Fragmentation Score is obtained by taking the average Fragmentation Score between each user and every other user.

Clustering Algorithm	**Representation**	**Abbreviation**
Louvain Community Detection	TF-IDF	baseline
Agglomerative Hierarchical Clustering	Sentence-BERT	AHC*SBERT
Agglomerative Hierarchical Clustering	Word embeddings	AHC*GloVe
Agglomerative Hierarchical Clustering	BoW	AHC*BoW
DB-Scan	Sentence-BERT	DB*SBERT
DB-Scan	Word embeddings	DB*GloVe
DB-Scan	BoW	DB*BoW

Table 4.4: Overview of all setups over which the Fragmentation Score is calculated, and their corresponding abbreviations.

The aggregated Fragmentation Score for each combination of the article representations and clustering algorithms is calculated. This results in 7 combinations: the baseline (TF-IDF*graph-based), and combinations of the three representation methods (BoW, word embeddings, and sentence embeddings) with the experimental clustering algorithms (agglomerative hierarchical clustering, and DB-scan). The combinations with their corresponding abbreviations are displayed in Table 4.4.

For each combination, three recommendation sets are generated (following Scenarios 1 to 3). This produces 21 sets of recommendations with predicted news story chains. The Fragmentation Score is calculated for each recommendation set. As there are 10 iterations of recommendation sets, where the randomly selected articles vary, the Fragmentation Score is calculated as the mean over the iterations. The standard deviation of all setups was smaller than 0.00, which indicates that the random selection of articles does not significantly affect the resulting Fragmentation Scores.

The comparison of the difference in Fragmentation Scores between the setups allows us to gain insight into how variations in the different clustering methods and representations affect the Fragmentation Scores. We could, for example, establish whether a high performance on the task of news story chain detection leads to a Fragmentation Score that is close to the gold score. Moreover, we might be able to determine what kind of mistakes in the clustering process results in an unreliable Fragmentation Score. A partition that tends to merge gold clusters might affect the Fragmentation Score to a different degree than a partition that contains many splits. An answer to these questions helps future users of the Fragmentation metric to make decisions in their pipeline that lead to a reliable measurement of Fragmentation.

Chapter 5

Results

This chapter reports the results of the two experiments that are outlined in Chapter 4. Section 5.1 reports the performance of different text representation methods and clustering algorithms on the task of news story chain detection, thus answering the first research question that inquires which text representation methods and clustering algorithms perform best on the task of newst story chain detection (see Section 1). Then, the resulting Fragmentation Scores for each setup based on scenarios that represent different reading preferences of users are presented in Section 5.1. This allows us to formulate an answer to the second research question, namely how variations in the chain detection system affect the Fragmentation score.

5.1 Clustering

This section reports and describes the results of the clustering setups. The performance is expressed in terms of the metrics described in Section 2.6: Homogeneity, Completeness, V-measure, Silhouette Score, and Davies-Bouldin Index. Section 5.1.1 provides a general overview of the results of each setup. Then, an error analysis is carried out for the three best-performing systems in Section 5.1.2. Recall from Section 4.4 that the setups under investigation contain the following text representations: Bag of Words (BoW); word embeddings (GloVe); sentence embeddings (SBERT), and the following clustering algorithms: agglomerative hierarchical clustering (AHC) and DB-Scan (DB) (as summarized in the previous section in Table 4.4 on page 36).

5.1.1 Overall Results

Table 5.1 displays the evaluation of the clustering methods. Almost all setups outperform the baseline by a large margin. The only exception is the DB*GloVe system, which scores very poorly on homogeneity. Table 5.2 presents the number of predicted clusters per setup. As can be seen, the baseline produces 79 clusters, which is a large deviation from the 7 true chains. A closer inspection of the clusters reveals that only 8 of the predicted clusters contain more than 1 article. These 8 clusters do not match the gold clusters, as they contain a variety of chains per cluster. There does not seem to be a pattern in the resulting partitions; most clusters contain articles from at least 5 gold chains.

According to the V-measure, the best-performing setup is AHC*SBERT. This comes as no surprise, as this representation method is specialized in capturing semantic sim-

Setup	**H** ↑	**C** ↑	**V** ↑	**S** ↑	**DBI** ↓
Baseline	0.166	0.156	0.161	-0.060	12.441
AHC*SBERT	**0.921**	**0.844**	**0.881**	0.290	1.933
AHC*GloVe	0.762	0.708	0.734	0.183	**1.913**
AHC*BoW	0.813	0.658	0.727	**0.413**	1.965
DB*SBERT	0.694	**0.872**	**0.773**	0.231	1.509
DB*GloVe	0.002	0.236	0.004	**0.390**	0.387
DB*BoW	**0.993**	0.283	0.441	0.213	**0.218**

Table 5.1: Evaluation of the different representation methods (Sentence-BERT, word embeddings, and Bag of Words) and clustering methods (agglomerative hierarchical clustering, and DB-Scan), and the baseline. The measures are abbreviated as follows: H (homogeneity), C (completeness), V (V-measure), S (Silhouette Score), and DBI (Davies-Bouldin Index). The arrow indicates whether a high or low score is more desirable.

Setup	**# Clusters**
Gold	7
Baseline	79
AHC*SBERT	9
AHC*GloVe	9
AHC*BoW	15
DB*SBERT	5
DB*GloVe	3
DB*BoW	868

Table 5.2: Number of clusters predicted by each system combination

ilarities between documents. DB*SBERT is the second-best setup, which illustrates the embeddings' ability to encode document similarity. The fact that sentence embeddings outperform every other combination of clustering algorithm and representation method indicates that advanced article representations are a more important consideration than the clustering algorithm. Although the V-measure is high for AHC*SBERT, this method produces 9 clusters, whereas there are 7 present in the gold data (see Table 5.2). The error analysis that follows this section sheds light on the contents of the superfluous clusters produced by this setup.

The worst-performing experimental setup (following the V-measure) is DB*GloVe. Both the homogeneity and completeness are low, indicating that the distribution of classes over clusters does not correspond to the gold story chains. As can be seen in Table 5.2, it produces only 3 clusters. Closer analysis of the predictions show that two of the clusters contain one article, while the remaining cluster contains all other articles. One of the articles displays the html code instead of the article text, and should have been removed in the preprocessing steps described in Chapter 3. The remaining article contains only one sentence, and is thus much shorter than most articles in the corpus. Notably, this setup achieves the highest Silhouette Score, which indicates that the predicted clusters contain objects that are similar to each other, but dissimilar to objects in the other clusters. This is only somewhat true in terms of the form of the articles: this setup identified outliers that differ from most articles by language and length, but it cannot distinguish between articles in terms of semantic similarity.

The second worst-performing setup is the DB*BoW. Although the homogeneity is high, the completeness is low. The high homogeneity can be explained by the large number of clusters that this approach produces, namely 868 (see Table 5.2). Most clusters contain only one article, which results in a high homogeneity because most clusters indeed contain data points from the same gold class. This, in turn, results in a low completeness because members of the same gold class are not collected in clusters, but are rather scattered. On the other hand, this approach achieves the lowest DBI, which should be an indication of well-separated clusters.

These results clearly reveal one of the drawbacks of the internal cluster validation metrics, namely that they tend to assign a better performance to density-based clustering algorithms, although their performance is lower than the hierarchical clustering methods in terms of V-measure. Comparison of the number of predicted clusters (see Table 5.2) shows that the V-measure assigns high values to setups that predict a number of clusters that is close to the gold clusters, while punishing predictions that deviate from the gold clusters.

The internal metrics can thus be said to give an inaccurate reflection of the clustering systems' performance. This should be kept in mind in future research on news story chain detection when the gold labels are not annotated. Evaluation of clusters without gold labels is difficult, and in this project, the internal evaluation metrics paint a picture of the setups' performance that is very different from the external evaluation metrics. Care should thus be taken to thoroughly inspect the quality of the predicted clusters, instead of only relying on internal metrics.

In summary, these results provide a convincing answer to the first research question, namely that based on the V-measure, the agglomerative hierarchical clustering algorithm outperforms the DB-Scan algorithm on the task of news story chain detection. Moreover, the Sentence-BERT representations achieve a higher V-measure than all other representation methods. The GloVe and BoW representations combined with agglomerative hierarchical clustering achieve a comparable V-measure, where the word embeddings slightly outperform the BoW representations. However, when combined with DB-Scan, the BoW representations outperform the GloVe word embeddings. Thus, in general, the ability of the representation method to capture semantic similarity is predictive of the system's performance. The only exception is DB*GloVe, as this setup is outperformed by the simpler text representation method of DB*BoW.

5.1.2 Error analysis

This error analysis compares the gold clusters to the predictions made by selected setups. This can be done by counting the number of times that a prediction differs from the gold label for each cluster. Because the predicted labels do not necessarily match the form of the gold labels (e.g. AHC*SBERT might assign the label 2 to the gold label of 4), this comparison only works for reasonably well-performing systems. In a poorly performing system, it is difficult to establish the majority label of a predicted cluster, which impedes the identification of individual errors. For this reason, the error analysis consists of an investigation of the mistakes that are made by the 3 best performing systems (AHC*SBERT, AHC*GloVe, and DB*SBERT) to the gold labels, as well as to each other. This thus includes a comparison between systems with the same representation method, but different clustering algorithms, and two systems with different representation methods but the same clustering algorithm.

The procedure of this analysis was comparing the predicted labels for individual gold clusters. More specifically, the number of errors in each cluster is counted for the approaches. The predicted majority label was identified for each setup, and the number of instances that deviate from this label were counted. This gives an indication of how the predicted clusters differ from the gold clusters, and allows us to answer the question of whether different approaches make the same kind of mistakes, and which classes are difficult to distinguish.

Table 5.3 displays the number of mistakes made by each setup under investigation, as well as the overlap in mistakes there are with the best-performing system AHC*SBERT (indicated between brackets after the total error count). Of the 58 mistakes that DB*SBERT made, 56 were also made by AHC*SBERT. AHC*GloVe incorrectly assigns 79 articles that are also assigned to the same cluster by AHC*SBERT. The large overlap indicates that the setups generally find the same articles difficult to cluster correctly, especially both SBERT systems. For all setups, the chain on the Wikileaks Trial seems the most difficult to identify correctly. A reason for this could be that two other chains (namely the Facebook Privacy Scandal and Equifax Breach) also report on events that are related to a data leak.

Gold label	**Size**	**# AHC*SBERT**	**# DB*SBERT**	**# AHC*GloVe**
2	167	9	5	3
4	163	4	4	38
5	152	55	36	54
6	83	2	2	39
7	99	20	9	39
8	126	1	1	20
9	241	9	1	45
Total		100	58 (56)	212 (79)

Table 5.3: The number of articles that are erroneously placed in a gold cluster. The number following the total between parentheses indicates the number of errors made by a setup that are also made by the best-performing AHC*SBERT. The gold labels correspond to the following news story chains: Ireland Abortion Vote (2); Facebook Privacy Scandal (4); Wikileaks Trials (5); Tunisia Protests (6); Ivory Coast Army Mutiny (7); Equifax Breach (8); and Brazil Dam Disaster (9).

Note that DB*SBERT makes the fewest mistakes when compared to the gold clusters, even though its V-measure is lower. This might be due to the number of merges this approach makes, as it only predicts 5 clusters. The completeness is relatively high, but the homogeneity is lower (see Table 5.1). This might explain why its overall performance seems high when only looking at the errors made when compared to the gold clusters: it assigns members of the same gold class to the same cluster (leading to a high completeness), but cannot always distinguish between classes and merges them (resulting in a lower homogeneity). In other words, most of its performance loss is due to the merging of classes, not the erroneous assignment of articles to chains they do not belong to. A closer inspection of the predicted clusters indicates that two clusters indeed fully contain multiple gold clusters. More specifically, DB*SBERT merges the chains about the Tunisia Protests and the Ivory Coast Army Mutiny into one cluster. Many of the articles on the Tunisia Protests report on the violence and looting that accompanied the protests, as well as the mass deployment of army forces to control the protests. A topical overlap can thus be found between the two merged chains. Moreover, DB*SBERT merges the chains about the Facebook Privacy Scandal, Wikileaks Trials, and Equifax Breach. Again, an overlap in topic can be found, as they all report on data leaks. It is thus not surprising that these chains are merged. This approach performs well on the chains about the Ireland Abortion Vote and the Brazil Dam Disaster; only a handful of articles were missing from the predicted clusters on these topics. The remaining predicted cluster, which is much smaller than the other 4, contains articles from 6 out of 7 gold clusters, and can thus be seen as a scattered collection of articles without a clear topical similarity.

A closer inspection of the clusters predicted by AHC*SBERT indicates that the predictions generally correspond well to the gold clusters. This setup produced a total of 9 clusters, which is 2 more than the gold clusters. Each topic in the evaluation set was assigned to its own cluster, indicating that AHC*SBERT manages to distinguish each news story chain reasonably well. For these 7 clusters, the number of articles it contains is close to the gold clusters. However, it is never exactly the same, which means that this setup tends to miss a handful of articles for each cluster. These missing articles are divided over the 2 remaining clusters. Further inspection of these clusters provides more insights. The first cluster contains a seemingly random collection of articles from all gold chains. There does not seem to be a clear explanation for the grouping of these articles, neither in terms of structure nor news outlet. The second cluster, however, contains articles that all originate from the same news outlet. The majority of these articles belong either to the Wikileaks Trials chain or the Ivory Coast Army Mutiny, which have no topical overlap. Notably, the other setups that are currently under investigation also assign one label to this group of articles. The investigation of the article texts points out that these articles report on the Russian invasion of Ukraine, although this topic should not be present in the HLGD. A plausible explanation of how this chain ended up in the dataset can be that the news outlet reused URLs for new articles. Since the data for this thesis was obtained by scraping the text from the URL, the updated text was extracted instead of the text that originally belonged to one of the news story chains. While both AHC*GloVe and DB*SBERT also assign the same label to these articles, only AHC*SBERT creates a new cluster to contain them. In other words, this method's evaluation scores would be even higher if this chain were annotated as such. This illustrates this setup's ability to distinguish between news story chains.

The clusters that AHC*GloVe provides are less clean than the two setups described above, which is reflected by the V-measure. The inspection of the clusters shows that despite the total number of 9 predicted clusters, this setup tends to merge gold clusters. The only clusters that are relatively homogeneous are those that contain articles on the Brazil Dam Disaster (although this topic is split over 2 clusters), the Ireland Abortion Vote, and, to a lesser extent, the Equifax Breach. The latter additionally contains a handful of articles from the technology-related story chains. The topics that are merged are Equifax Breach and the Facebook Privacy Scandal, the Tunisia Protests and Ivory Coast Army Mutiny, and the Wikileaks Trials and Ivory Coast Army Mutiny. Moreover, AHC*GloVe produces two clusters that contain several articles without clear topic coherence.

5.2 Fragmentation

This section reports the results of the experiment that aims to investigate the effect of mistakes is the news story chain detection system on the resulting Fragmentation Score. For instance, consider a scenario in which one class is spread over 5 clusters. Most articles that are read belong to a single cluster, although the specific cluster may differ between users. The pipeline would generate a high Fragmentation Score, because the readers are exposed to different story chains. However, in reality, the Fragmentation Score should be lower, as they all read articles from the same story chain. Conversely, when classes are wrongly merged into the same cluster, the Fragmentation score will be low, thus mistakenly painting a situation in which readers are exposed to the same news story chains.

The Fragmentation metric takes ordered list of recommended news articles as input, and calculates the overlap in news story chains between users. The recommendation sets for a group of simulated users were constructed on the basis of three scenarios. According to the gold labels, Scenario 1 should lead to low Fragmentation, Scenario 2 to high Fragmentation, and Scenario 3 to a balanced Fragmentation. Table 5.4 displays the Fragmentation Scores for each scenario per setup, as well as the variation in the Fragmentation Score across scenarios. As the generation of the recommendation sets contains a random selection of articles, 10 distinct sets were generated. It was established that the resulting standard deviation of Fragmentation Scores between the different selections is lower than 0.001, which can be regarded negligible.

Setup	**Scen. 1** ↓	**Scen. 2** ↑	**Scen. 3**	**Variation**
Gold	0.00	0.85	0.58	0.85
Baseline	0.67	0.73	0.70	0.06
AHC*SBERT	0.31	0.87	0.64	0.56
AHC*GloVe	0.38	0.84	0.63	0.46
AHC*BoW	0.62	0.85	0.63	0.23
DB*SBERT	0.16	0.74	0.48	0.58
DB*GloVe	0.01	0.01	0.00	0.01
DB*BoW	0.99	0.99	0.99	0.00

Table 5.4: Fragmentation Scores for each setup per scenario

The first row of Table 5.4 displays the Fragmentation Scores that are produced by the gold articles, to which the scores following each experimental system can be compared. The scores should not be interpreted individually per scenario, the focus should rather be on the pattern that the Fragmentation Scores reveal across scenarios. An accurate Fragmentation Score would display a high variability across scenarios, whereas a meaningless score stays stable in different scenarios. For example, the DB*GloVe setup achieves an accurate Fragmentation Score in Scenario 1, but it provides the same score for Scenarios 2 and 3. On the other hand, the Fragmentation patterns of the best-performing clustering systems (AHC*SBERT, AHC*GloVe and DB*SBERT) also display the largest variability. These systems manage to capture a change in Fragmentation based on differences in the recommendation sets.

The largest variation in the Fragmentation Score across scenarios is found in DB*SBERT, which has a V-measure of 0.773 on the news chain detection task. It displays only slightly more variance than the best-performing system AHC*SBERT, although its V-measure is considerably higher (namely 0.881). It appears that the well-performing systems generally tend to report a Fragmentation Score that is higher than the gold score. This is especially true for Scenario 1, and to a lesser extend to Scenario 3. In other words, the variation is mostly inhibited by the difficulty to detect a low Fragmentation. The larger variation in DB*SBERT can be explained by the lower number of clusters it generates, as it produces 5 clusters while AHC*SBERT produces 9 clusters. Fewer splits will logically result in a lower Fragmentation Score, as the chance that two articles originate from the same story chain is bigger when there are fewer clusters.

The baseline, DB*GloVe, and DB*BoW perform poorly in terms of clustering, and also display a very small variation in Fragmentation Scores across scenarios. These setups can thus be regarded as unsuited for the current task. Of the remaining systems, AHC*BoW has the lowest degree of variation, as the Fragmentation Score will tend to be high in any scenario due to the large number of splits. The variation in AHC*GloVe is considerably higher, but not as high as in both SBERT setups. The accuracy with which both AHC*SBERT and DB*SBERT approach the gold Fragmentation Scores, and their ability to detect story chains, make these setups favorable approaches for this task.

All in all, it seems that a high performance on news story chain detection is a good indicator for the reliability of the resulting Fragmentation Score. The best-performing systems in terms of clustering, AHC*SBERT and DB*SBERT, are also found to have the most accurate Fragmentation Score across scenarios. Systems that produce more splits than necessary will deviate towards a lower Fragmentation, whereas merges lead to a higher Fragmentation Score. If too many splits or merges are made, as is the case with DB*GloVe and DB*BoW, the Fragmentation Score becomes unreliable, as it remains stable over different scenarios, thus giving a distorted view of its performance.

5.3 Summary of Results

The main findings of this thesis are summarized below. Firstly, the results of the first experiments are displayed. This answers the first research question and part of the sub-question, namely how various text representation methods and clustering approaches perform on the task of news story chain detection. Secondly, the second research question can be answered, namely how the Fragmentation Score is influenced by variations in the chain detection system.

1. Clustering

    - The baseline is outperformed by all but one experimental system (namely DB*GloVe);
    - Contextualized sentence embeddings lead to the highest performance on news story chain detection. AHC*SBERT achieves the highest performance, followed by DB*SBERT;
    - The Agglomerative Hierarchical Clustering algorithm outperforms DB-Scan for all representation method;
    - The internal cluster validation metrics tend to paint an unrealistic picture of the performance.

2. Fragmentation

    - The difference between the highest and lowest Fragmentation Score of a setup is a good indicator of the score's reliability, since the Fragmentation Scores should vary between scenarios;
    - The SBERT setups display the highest variation in Fragmentation Scores, while also being close to the gold Fragmentation Score;
    - The number of clusters that a setup produces has a strong influence on the resulting Fragmentation Score: more splits lead to a higher Fragmentation, whereas merges result in a lower Fragmentation Score, regardless of the type of news recommendations.

3. General

    - AHC*SBERT achieves the highest performance on clustering, and the resulting Fragmentation Score is close to the gold score. This system can thus be regarded as the best setup that was tested in this thesis.

Chapter 6

Discussion and Conclusion

In this chapter, the results that are reported in Chapter 5 are discussed in the light of several limitations. Additionally, directions for future work are reported, as well as a summary of the recommendations that can be made based on the findings of the clustering experiments.

6.1 Discussion

The aim of this thesis is to investigate how the Fragmentation of a set of personalized news recommendations is affected by errors that are made in a pipeline that identifies news story chains. The detection of news story chain is necessary because the Fragmentation score calculates the overlap in exposure to chains between users. Finding a suitable corpus for the current task is challenging, because it should contain annotations for news story chains as well as user reading behavior. The latter is required to generate the sets of news recommendations over which the Fragmentation Score is calculated. To our knowledge, no such corpus has yet been developed. The HeadLine Grouping Dataset (HLGD) is suitable for the task of news story chain detection, although it has some characteristics that do not fully reflect a realistic news article corpus. The necessity to simulate user reading profiles instead of using a corpus with real user behaviors adds more artificiality to the experiments. This section discusses the findings of this thesis in the light of these limitations. Firstly, limitations related to the corpus are addressed, followed by a discussion of the simulated reader scenarios. The section concludes by summarizing the recommendations that follow from this thesis. Finally, an overview of the general conclusions that can be drawn from the results is presented.

Limitations of the Corpus

An inherent challenge of experiments that involve annotated data is establishing how well the results translate to the real world. For the current dataset, this is especially true, as it contains several characteristics that are not fully reflective of reality. This section discusses the most important limitations of the HLGD dataset.

Firstly, the time span of the news story chains in the corpus ranges from 18 days to 10 years. Following the definition of news story chains as formulated by Nicholls and Bright (2019), articles within a story chain are usually published within a proximity of no more than three days. Although longer spans can occur for more impactful or controversial events, it is an exception rather than a rule. Thus, a more realistic corpus would reflect this pattern and thus contain more story chains that comprise shorter time spans.

Secondly, the HLGD does not include singleton story chains, even though these undoubtedly will be present in a more realistic corpus. This makes it difficult to draw conclusions from the results that translate to a realistic use-case scenario. For example, a news outlet that generates personalized recommendations to their users might want to investigate the degree of Fragmentation in the output of their system. Their corpus would consist of various news story chains as well as single-event articles. More research should be done to establish the ability of the clustering algorithms to identify these singleton clusters. The DB-Scan algorithm has been found to deal better with variations in cluster sizes, and might thus perform better on a corpus that contains singleton news story chains compared to hierarchical clustering. A follow-up study could compare the performance of different clustering algorithms on a corpus containing both singleton articles and news story chains to gain insight into this question.

Additionally, it would be interesting to see how the tested clustering methods can deal with more fine-grained story chains. Although the current dataset contains a few story chains that are somewhat related to each other in terms of the broader topic, there are no sub-events of the same overarching story present. To fully assess an algorithm's ability to identify news story chains, the task should be performed on a dataset that contains story chains that report on different aspects of the same topic. For example, it could contain articles about a number of soccer matches, where the articles that report on each match represent an individual news story chain. It would be insightful to find out how well the best-performing system can identify individual matches from a pool of soccer-related articles.

It should be noted that the true news story chains would be unknown in a scenario that follows reality more closely. This would make the evaluation of the clusters difficult, because the internal validation metrics (i.e. the Silhouette Score and the Davies-Bouldin Index) that would be necessary when labels are absent turned out to produce inaccurate scores. However, the performance of the current pipeline could be verified by evaluating it on other corpora with annotated news story chains, such as the Business Energy News dataset (Gedikli et al., 2021), or the annotated pairs from Trilling and van Hoof (2020). Both datasets have shortcomings; the Business Energy News dataset is domain specific, and Trilling and van Hoof (2020) made a selection of a small portion of the data for validation, which does not lead to an independent evaluation. Despite their drawbacks, evaluation on these datasets can yield complementary insights on the performance of the setup that is proposed in this thesis.

All in all, the choice of the corpus led to experiments that are rather artificial. The exceptionally long time range of the news story chains in the corpus, the absence of singleton clusters, as well as the relatively coarse-grained story chains increase the artificiality of the experiments. This line of research can thus benefit greatly from a rich dataset that solves these problems. Especially the inclusion of different levels of granularity preferably structured as overarching topics that contain smaller news chains would yield more in-depth insights into the ability of clustering algorithms to identify the chains. Furthermore, it should include information on user reading behaviors such that realistic recommendation sets can easily be generated. The following section elaborates on this point.

The Simulated Reader Scenarios

Due to the unavailability of a corpus that combines news story chain annotations and user interactions with news articles, we had to design simulated recommendation sets. The first option we considered consisted of transferring data on user interactions with news articles from another corpus to the HLGD articles by matching the articles in terms of similarity. Recommendations could then be generated by an existing news recommender. However, due to the low degree of similarity between the articles, and the fact that the quality of these steps cannot accurately be evaluated, this idea was abandoned. Instead, we opted for simulating the recommendation sets based on different assumptions of reader behaviors. This resulted in three scenarios with varying degrees of expected Fragmentation. Despite efforts to include variation between users in one of the scenarios, the resulting scenarios are rather simplistic and do not represent realistic reader behaviors well. For example, Ohlsson et al. (2017) found that a variety of individual factors influence reading behavior in online news contexts. These factors include class, age, gender, political interests, and cognitive abilities. A realistic set of news recommendations should include more diverse reader profiles to account for these variations.

Nevertheless, it should be noted that an artificial setting is a useful first approach to investigating the effect of the different clustering systems affect the Fragmentation Score, since it allowed us to control for variations. By simulating recommendations that lead to, for instance, low Fragmentation Scores, it was easy to identify that the use of Agglomerative Hierarchical Clustering results in more accurate clusters as well as Fragmentation Scores. The degree to which these results generalize to real-life scenarios where more complexity is present, both in terms of user interactions and granularity of news story chains, remains a topic for further research.

In short, the main limitations of this project concern the artificiality of the dataset and the experiments. More work could be done on how well the findings generalize to the other datasets that are available for this task. Furthermore, the development of a dataset that is enriched with user interactions, shorter spans, single-article news chains and fine-grained story chains would help this line of research move forward.

Recommendations

This thesis found that the SBERT sentence embeddings combined with the agglomerative hierarchical clustering algorithm performs best at the task of detecting news story chains. With a V-measure of 0.881, it manages to assign the majority of articles to the correct news story chain. Most care should be taken for selecting a text representation method, as more sophisticated representations are a strong indication of the quality of the resulting clusters. In turn, well-formed clusters lead to a reliable Fragmentation Score, even when the clusters contain some errors. When external evaluation metrics are unusable due to the absence of true class labels, the variation in Fragmentation Scores within clustering setups across different news recommendation sets with different distributions of news story chains is a good indication of its reliability. Moreover, a low variation in the Fragmentation Score across scenarios can be used as an indication of inaccurate clusters. It is a strong clue that a clustering systems performs poorly if the Fragmentation Score stays relatively stable even though the recommendation sets contain drastically different distributions of news story chains. The variation in the Fragmentation Score can thus be used as a quick cluster validation method: if the score remains stable across different random recommendation sets, it would be a good idea to inspect the accuracy of the clusters.

Naturally, it was not possible to include all text representation or clustering methods that potentially perform well on news story chain detection. Future work could investigate how various other clustering setups manage the task. A more extensive comparison between the hierarchical clustering algorithm and the graph-based methods that were previously applied to this task may yield interesting results. An extension of this thesis could combine the SBERT sentence embeddings with the graph-based Louvain algorithm that was used as a baseline and extensively compare the strengths and weaknesses of both methods in the context of news story chain detection. Currently, the sentence embeddings are not combined with the graph-based clustering algorithm due to time restrictions. Investigating this combination allows for better comparison to previous approaches of news chain detection, because a graph-based method is often used for this task. It might be the case that this class of algorithm is adept at finding fine-grained clusters or singleton clusters when combined with sophisticated text representations.

6.2 Conclusion

This thesis aims to expand the technical implementation of the Fragmentation metric, developed by Vrijenhoek et al. (2021). The Fragmentation Score measures the overlap in the news story chains that are present in sets of news articles that are recommended to users of a personalized news recommendation system. This requires a system that can automatically detect news story chains in a set of articles. A common approach to this task is by means of clustering, which is what the first part of the experiments is concerned with. The first research question was formulated as “How do various clustering approaches perform on the task of news story chain detection?”. To answer this question, the agglomerative hierarchical clustering and DB-Scan algorithms are compared in terms of performance. Additionally, the experiment includes various text representation methods, namely BoW representations, word embeddings, and sentence embeddings. The second set of experiments investigates how the Fragmentation Score is influenced by variations in the news story chain detection system, both in terms of the choice of the clustering algorithm and the text representation method. This was done by simulating three scenarios in which sets of news recommendations have varying degrees of Fragmentation, and comparing the resulting Fragmentation Scores of the experimental systems to the gold scores.

To our knowledge, this thesis is the first project to extensively evaluate the performance of different suitable setups on the task of news story chain detection while providing the Fragmentation Scores that follow. We found that the sentence embeddings combined with agglomerative hierarchical clustering overwhelmingly outperform the baseline as well as other combinations on the task of news story chain detection. Moreover, as the second-best performance was achieved with the sentence embeddings and DB-Scan, it can be concluded that the use of sophisticated text representation methods, such as sentence embeddings from a contextualized language model, are the most important prerequisite for the success of a news story chain detection system. It was also found that the degree to which the resulting Fragmentation Scores are indeed indicative of the true Fragmentation in the recommendations are highly dependent on the performance on news story chain detection. The best-performing systems lead to a

Fragmentation Score that mimics the gold score relatively well across the constructed scenarios. Conversely, the poor-performing clustering systems generate a pattern of Fragmentation Scores that are stable across scenarios; they are thus not able to capture the variance that should result from the different recommendation sets. In other words, the variation in Fragmentation Scores across scenarios can be used as an initial cluster validation method.

The main limitation of this thesis is the fact that the corpus that was used does not fully reflect a realistic set of news articles: the news story chains span a longer time than usual, they do not contain articles that are not part of a story chain, and, most importantly, they are generally dissimilar to each other. In a realistic scenario, multiple news story chains may relate to the same broader topic, which makes the task of distinguishing them more challenging. This calls for the need of a larger dataset with annotations of more fine-grained news story chains, as well as annotations for user interactions with the articles. Despite the artificial nature of the data and experiments, the results convincingly show that the current implementations of clustering that are used to calculate Fragmentation (e.g. in Vrijenhoek et al. (2021)) are not sufficiently reliable. The pipeline that is proposed in this thesis yields more accurate approximations of the Fragmentation Score.

Bibliography

L. M. Abualigah, A. T. Khader, and E. S. Hanandeh. Hybrid clustering analysis using improved krill herd algorithm. *Applied Intelligence*, 48(11):4047–4071, 2018.

C. C. Aggarwal and C. Zhai. A survey of text clustering algorithms. In *Mining text data*, pages 77–128. Springer, 2012.

O. Arbelaitz, I. Gurrutxaga, J. Muguerza, J. M. Pérez, and I. Perona. An extensive comparative study of cluster validity indices. *Pattern recognition*, 46(1):243–256, 2013.

K. Babić, S. Martinčić-Ipšić, and A. Meštrović. Survey of neural text representation models. *Information*, 11(11):511, 2020.

A. Barbaresi. Trafilatura: A Web Scraping Library and Command-Line Tool for Text Discovery and Extraction. In *Proceedings of the Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 122–131. Association for Computational Linguistics, 2021. URL `https://aclanthology.org/2021.acl-demo.15`.

P. Berkhin. A survey of clustering data mining techniques. In *Grouping multidimensional data*, pages 25–71. Springer, 2006.

A. Bernstein, C. de Vreese, N. Helberger, W. Schulz, K. Zweig, C. Baden, M. A. Beam, M. P. Hauer, L. Heitz, P. Jürgens10, et al. Diversity in news recommendation. *Perspectives*, 24:29, 2019.

V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008, 2008.

A. Bouguettaya, Q. Yu, X. Liu, X. Zhou, and A. Song. Efficient agglomerative hierarchical clustering. *Expert Systems with Applications*, 42(5):2785–2797, 2015.

J. Boumans, D. Trilling, R. Vliegenthart, and H. Boomgaarden. The agency makes the (online) news world go round: The impact of news agency content on print and online news. *International Journal of Communication*, 12:22, 2018.

R. Burke. Hybrid recommender systems: Survey and experiments. *User modeling and user-adapted interaction*, 12(4):331–370, 2002.

T. Caliński and J. Harabasz. A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods*, 3(1):1–27, 1974.

Z. Chen and H. Ji. Graph-based clustering for computational linguistics: A survey. In *Proceedings of TextGraphs-5-2010 Workshop on Graph-based Methods for Natural Language Processing*, pages 1–9, 2010.

C. G. Christians, T. Glasser, D. McQuail, K. Nordenstreng, and R. A. White. *Normative theories of the media: Journalism in democratic societies*. University of Illinois Press, 2010.

D. L. Davies and D. W. Bouldin. A cluster separation measure. *IEEE transactions on pattern analysis and machine intelligence*, (2):224–227, 1979.

R. C. de Amorim. Feature relevance in ward's hierarchical clustering using the l p norm. *Journal of Classification*, 32(1):46–62, 2015.

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

M. Ester, H.-P. Kriegel, J. Sander, X. Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd*, volume 96, pages 226–231, 1996.

M. M. Ferree, W. A. Gamson, J. Gerhards, and D. Rucht. Four models of the public sphere in modern democracies. *Theory and society*, 31(3):289–324, 2002.

F. Gedikli, A. S. Novo, and D. Jannach. Semi-automated identification of news story chains: A new dataset and entity-based labeling method. 2021.

Z. S. Harris. Distributional structure. *Word*, 10(2-3):146–162, 1954.

J. He, A.-H. Tan, C.-L. Tan, and S.-Y. Sung. On quantitative evaluation of clustering systems. In *Clustering and information retrieval*, pages 105–133. Springer, 2004.

N. Helberger. On the democratic role of news recommenders. *Digital Journalism*, 7(8): 993–1012, 2019.

D. Held. Models of democracy. 2006.

K. S. Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 1972.

K. Karppinen. Uses of democratic theory in media and communication studies. *Observatorio*, 7(3):1–17, 2013. ISSN 1646-5954.

M. Kunaver and T. Požrl. Diversity in recommender systems–a survey. *Knowledge-based systems*, 123:154–162, 2017.

M. Kusner, Y. Sun, N. Kolkin, and K. Weinberger. From word embeddings to document distances. In *International conference on machine learning*, pages 957–966. PMLR, 2015.

P. Laban, L. Bandarkar, and M. A. Hearst. News headline grouping as a challenging nlu task. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3186–3198, 2021.

A. Lancichinetti and S. Fortunato. Community detection algorithms: a comparative analysis. *Physical review E*, 80(5):056117, 2009.

O. Levy, Y. Goldberg, and I. Dagan. Improving distributional similarity with lessons learned from word embeddings. *Transactions of the association for computational linguistics*, 3:211–225, 2015.

M. Li and L. Wang. A survey on personalized news recommendation technology. *IEEE Access*, 7:145861–145879, 2019.

Y. Liu, Z. Li, H. Xiong, X. Gao, and J. Wu. Understanding of internal clustering validation measures. In *2010 IEEE international conference on data mining*, pages 911–916. IEEE, 2010.

B. Manin. On legitimacy and political deliberation. *Political theory*, 15(3):338–368, 1987.

S. M. McNee, J. Riedl, and J. A. Konstan. Being accurate is not enough: how accuracy metrics have hurt recommender systems. In *CHI'06 extended abstracts on Human factors in computing systems*, pages 1097–1101, 2006.

P. Melville, R. J. Mooney, R. Nagarajan, et al. Content-boosted collaborative filtering for improved recommendations. *Aaai/iaai*, 23:187–192, 2002.

T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *International Conference on Learning Representations*, 2013a.

T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26, 2013b.

T. Mikolov, W.-t. Yih, and G. Zweig. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 conference of the north american chapter of the association for computational linguistics: Human language technologies*, pages 746–751, 2013c.

F. Murtagh and P. Contreras. Algorithms for hierarchical clustering: an overview. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2(1):86–97, 2012.

T. Nicholls and J. Bright. Understanding news story chains using information retrieval and network clustering techniques. *Communication methods and measures*, 13(1): 43–59, 2019.

J. Ohlsson, J. Lindell, and S. Arkhede. A matter of cultural distinction: News consumption in the online media landscape. *European Journal of Communication*, 32 (2):116–130, 2017.

J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

N. Reimers and I. Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019.

A. Rosenberg and J. Hirschberg. V-measure: A conditional entropy-based external cluster evaluation measure. In *Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)*, pages 410–420, 2007.

P. J. Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.

K. Scott. You won't believe what's in this paper! clickbait, relevance and the curiosity gap. *Journal of pragmatics*, 175:53–66, 2021.

S. Scott and S. Matwin. Text classification using wordnet hypernyms. In *Usage of WordNet in natural language processing systems*, 1998.

J. Strömbäck. In search of a standard: Four models of democracy and their normative implications for journalism. *Journalism studies*, 6(3):331–345, 2005.

P. B. Thorat, R. Goudar, and S. Barve. Survey on collaborative filtering, content-based filtering and hybrid recommendation system. *International Journal of Computer Applications*, 110(4):31–36, 2015.

D. Trilling and K. Schoenbach. Patterns of news consumption in austria: how fragmented are they? *International Journal of Communication*, 7:25, 2013.

D. Trilling and M. van Hoof. Between article and topic: News events as level of analysis and their computational identification. *Digital Journalism*, 8(10):1317–1337, 2020.

S. Vrijenhoek, M. Kaya, N. Metoui, J. Möller, D. Odijk, and N. Helberger. Recommenders with a mission: assessing diversity in news recommendations. In *Proceedings of the 2021 Conference on Human Information Interaction and Retrieval*, pages 173–183, 2021.

W. Webber, A. Moffat, and J. Zobel. A similarity measure for indefinite rankings. *ACM Transactions on Information Systems (TOIS)*, 28(4):1–38, 2010.

I. M. Young. *Six Communication and the Other: Beyond Deliberative Democracy*, pages 120–136. Princeton University Press, 2021. doi: doi:10.1515/9780691234168-007. URL `https://doi.org/10.1515/9780691234168-007`.

W. Zhang, T. Yoshida, and X. Tang. A comparative study of tf* idf, lsi and multi-words for text classification. *Expert Systems with Applications*, 38(3):2758–2765, 2011.

L. Zheng, L. Li, W. Hong, and T. Li. Penetrate: Personalized news recommendation using ensemble hierarchical clustering. *Expert Systems with Applications*, 40(6):2127–2136, 2013.