Università di Firenze, Università di Perugia, INdAM consorziate nel CIAFM

DOTTORATO DI RICERCA
IN MATEMATICA, INFORMATICA, STATISTICA

CURRICULUM IN STATISTICA
CICLO XXXV

Sede amministrativa Università degli Studi di Firenze
Coordinatore Prof. Paolo Salani

Extracting knowledge from text news: A systematic evaluation of network-based topic detection

Settore Scientifico Disciplinare SECS-S/05

Dottorando:
Carla Galluccio

Tutor
Prof.ssa Alessandra Petrucci

Coordinatore
Prof. Matteo Focardi

Anni 2019/2022

Acknowledgements

The author acknowledge the financial support provided by the "Dipartimenti Eccellenti 2018-2022" ministerial funds. This work has been partly developed during the Ph.D. visiting period at Uppsala University, Sweden.

Part of the chapters' content is included in the papers:

Galluccio, C., Crescenzi, F. & Petrucci, A. (2021). "The Italian Newspapers' Narrative on Distance Learning during Covid-19 Pandemic", Statistica Applicata - Italian Journal of Applied Statistics, 33 (2), pp. 107 - 122;

Galluccio, C., Magnani, M., Vega, D., Ragozini, G. & Petrucci, A. (2023). "Robustness and sensitivity of network-based topic detection". In: Cherifi, H., Nunzio Mantegna, R., Rocha, L.M., Cherifi, C, Micciche, S. (eds.). Complex Networks & Their Applications XI: Proceedings of The Eleventh International Conference on Complex Networks and their Applications (pp. 259 - 270). Springer Cham, Switzerland.

Summary

In the context of textual analysis, network-based procedures for topic detection are gaining attention, also as an alternative to classical topic models. These procedures are based on the idea that documents can be represented as word co-occurrence networks, where topics are defined as groups of strongly connected words. Although many works have used network-based procedures for topic detection, there is a lack of systematic analysis of how different design choices, such as building the word co-occurrence matrix and selecting the community detection algorithm, affect the final results in terms of detected topics. Another unexplored question about network-based topic detection concerns its relationship with classical topic models, such as the Latent Dirichlet Allocation (LDA) model. Therefore, this thesis aims to address these questions by developing a deeper understanding of optimal design choices for network-based procedures for topic detection, showing how and to what extent the choices made during the design phase affect the results, and contextually comparing these procedures with classical topic models.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Information society and the role of big data

The concept of "information society" refers to a society in which the production, distribution and use of information represent essential economic and cultural activities (Mansell, 2010)

In an information society, information and communication technologies (ICTs) play a fundamental role in the way people communicate, work and access information and resources (Van Dijk, 2020). As stated in Floridi (2014), the evolution of ICTs is modifying the way we lead our lives and interpret the world in terms of information, proposing a novel perception of the space and time in which we live.

The dependence of information societies on the evolution of digital devices for the creation, conservation and transmission of data, in

quantities never experienced before, would have sanctioned humanity's entry into an "hypershistoric" era, the age of the zettabyte[1], in which a sufficient number of data "to fill all US libraries eight times over" (Floridi, 2012, p.435) are generated daily, such as to force the introduction of a neologism, "big data", to restore the knowledge of the amount of information produced (G. Galluccio, 2021).

The rise of the information society and the emergence of big data have had significant impacts on various aspects of society, including the economy, politics and culture. In particular, big data have the potential to facilitate the creation of new products and services, improve the efficiency and effectiveness of organisations and inform policy decisions (Gandomi & Haider, 2015).

One of the main ways in which big data are being used in the information society is for business and economic purposes. For example, big data can be used to improve customer targeting and optimise supply chains. By analysing large datasets, businesses can identify trends, patterns and relationships that were previously invisible, using this information successively to make better decisions and gain a competitive advantage (Gandomi & Haider, 2015; McAfee, Brynjolfsson, Davenport, Patil, & Barton, 2012).

Big data are also used in the public sector to improve the delivery of services and to inform policy decisions. Governments can use big data e.g. to monitor and predict the spread of diseases, optimise transportation systems and improve public service efficiency. The analysis of this vast amount of data allows governments to identify patterns and trends that can inform policy decisions and help improve citizens' lives (Mayer-Schönberger & Cukier, 2013).

[1]One zettabyte is equivalent to a trillion gigabytes.

In addition to its economic and policy applications, big data broadly impact also culture and society. For example, big data are used to study and understand social media interactions, political discourse and cultural trends, allowing researchers to identify patterns that can provide insights into social and cultural phenomena, such as the spread of ideas, the formation of social networks and the influence of media (Boyd & Crawford, 2012).

However, one of the main challenges in using big data is that they often include a combination of structured and unstructured data. Structured data refer to data stored in databases or spreadsheets, organised according to strict schemes and tables. Structured data are typically easy to process and analyse, as they follow a well-defined format and can be accessed and manipulated using relational information management models. Unstructured data, on the other hand, refer to data with no predetermined, systematic structure: they are stored without any patterns. Examples of unstructured data include text documents, emails, social media posts and images. This kind of data can not be easily processed or analysed using traditional methods, but they must be transformed to give them a structure. Therefore, analysing and gaining insights from big data is particularly challenging as they require specialised tools and techniques for handling both structured and unstructured data (Eberendu, 2016).

In summary, the complexity of contemporary social, economic and environmental dynamics makes the effectiveness of any human activity increasingly subordinated to management strategies of information flows capable of supporting decision-making processes, within which information assumes, therefore, an ethical, political and economic value. In other words, in information societies information takes on an epistemological value, becoming at the same time the main substance

and key to understanding the world. In this context, data play a fundamental role and consequently the process of extracting knowledge from them.

In such a defined context, it therefore becomes essential to recall the concept of Knowledge Discovery (both from structured and unstructured data) to determine what can be considered "knowledge" in this flood of data and information.

1.2 From data to knowledge

Knowledge Discovery (KD) is defined as: "[...] *the nontrivial extraction of implicit, previously unknown, and potentially useful information from data*" (Frawley, Piatetsky-Shapiro, & Matheus, 1992, p.58). This definition emphasises the "discovery of knowledge" process that occurs by extracting information from data, regardless of the specific nature of the data (Dulli, Polpettini, & Trotta, 2004). The KD process is closely related to the concept of pattern, which can be defined as follows: "*given a set of facts (data) F, a language L, and some measure of certainty C, we define a pattern as a statement S in L that describes relationships among a subset F_S of F with a certainty c, such that S is simpler (in some sense) than the enumeration of all facts in F_S*" (Frawley et al., 1992, p.58).

Considered a set of data, a KD process consists of searching for patterns describing the relationships between the elements that constitute a subset of the data set (Cios, Swiniarski, Pedrycz, & Kurgan, 2007; Frawley et al., 1992). Although it might be possible to discover and extract several types of data relationships, the patterns considered in a KD context are those expressed in a high-level language. This

expression in the field of computer science refers to programming languages that show significant levels of abstraction. This means these programming languages are designed to be easily understood by humans. Hence, they are characterised by typical elements of natural language. To be used by a computer, programs written in a high-level language must be translated or interpreted by other programs. The extracted patterns can be directed and used by both users and other programs, representing, in the latter case, additional inputs for subsequent processing (Frawley et al., 1992).

However, the patterns that can be extracted from a subset of data are potentially infinite. Still, only those considered interesting and reliable from the user's point of view are considered knowledge. Patterns are interesting when they are new, useful, and non-trivial to compute. Knowledge is useful when it can help to achieve system or user goals successfully. Patterns completely unrelated to the targets of interest are of little use and, therefore, not considered knowledge.

Originality and usefulness alone, however, are not sufficient to qualify a pattern as knowledge. Most databases contain many new and useful patterns, but they are not considered knowledge because they are trivial to compute. For patterns to be considered as non-trivial to compute, it is necessary that the system, through them, does more than blindly process elementary statistics. In fact, according to the observations behind what has been said so far, the results of the direct processing of simple statistics are readily available from the user's database, thus not requiring the need to extract and process patterns to obtain them.

A discovery system, i.e. a system centred on the discovery of knowledge, must therefore be capable of deciding which processes to carry out and whether the results obtained from these processes are interesting enough to be considered knowledge. Another way of seeing

the notion of non-triviality is connected to the idea that a discovery system must possess a certain degree of autonomy in data processing and evaluating results (Cios et al., 2007; Frawley et al., 1992).

As regards the reliability of the patterns, the representation and description of the degree of reliability are essential in determining the trustworthiness of discovery for the user. Nevertheless, certainty is rarely a component of the discovery of knowledge. Despite that, relying on a certain degree of reliability is essential because, without it, the choice of only a circumscribed number of patterns would be unwarranted, and consequently, it would be impossible to consider them as knowledge. The concept of reliability requires some basic elements, such as the integrity of the data, the sample size and, when possible, a reference knowledge domain (Frawley et al., 1992).

Patterns, therefore, are interesting when they are new, useful, and non-trivial to compute, and they are reliable when applied to all data they turn out to be somewhat certain. The possibility of taking a new pattern into account depends on the considered reference framework, which can be defined accordingly to the aim of the knowledge process carried out by the system or the user (Dulli et al., 2004).

Finally, another fundamental element in this context is that a computer should efficiently implement a discovery process. Here, the concept of efficiency refers especially to the algorithms' time and space consumption when used in the model implementation phase.

Therefore, it is possible to state that the KD process is made up of four main components: "high-level language", meaning that knowledge discovery is represented by a high-level language that does not necessarily need to be used directly by humans, although it requires that its expressions should be understandable also by them; "accuracy", meaning that findings must accurately describe the contents of the database, where measures of reliability express the extent to which this

representation is flawed; "interesting outcomes", meaning that knowledge discovery turns out to be interesting with respect to user-defined goals, thus implying that the patterns are novel and potentially useful and that the discovery process is non-trivial; "efficiency", meaning that the discovery process is efficient, a characteristic represented by the predictability and the acceptability of the execution time of the algorithms used on large databases of interest (Frawley et al., 1992).

To conclude, it is possible to define the discovered knowledge as the result obtained from the elaboration of a program that analyses the set of observations contained in a database and extracts patterns (Frawley et al., 1992). This last definition represents the intersection between KD and Knowledge Discovery in Database (KDD; for further information, see Cios et al., 2007; Fayyad, Piatetsky-Shapiro, & Smyth, 1996; Frawley et al., 1992). Indeed, while KD is defined as a paradigm that refers to the overall process of extracting knowledge from data regardless of their nature, KDD refers specifically to the extraction of information from structured databases (Dulli et al., 2004). Instead, when applied to textual databases, the KD process can be defined as Knowledge Discovery from Text (KDT).

1.2.1 Knowledge Discovery from Text

As regards Knowledge Discovery from Text (KDT), it can be defined as the non-trivial process of identifying valid, potentially useful and finally understandable patterns from textual data (Feldman & Dagan, 1995). Most previous works in knowledge discovery focused on structured databases and extracting information from them. However, nowadays, a large portion of the available information is not kept in structured

databases but rather in various formats, such as collections of textual documents drawn from different sources (Feldman & Dagan, 1995).

As an example, the International Data Corporation (IDC), a global provider of market intelligence, consulting and IT and digital innovation events, in a 2018 report sponsored by EMC Corporation (leading IT infrastructure manufacturer, storage, business intelligence and virtualisation, acquired by Dell in 2015) predicted that the volume of data would be increased by 175 zettabytes by 2025, mainly determined by the growing number of devices and sensors[2]. Another example is given by the Cisco Visual Networking Index for the years 2017-2022, based on which it was predicted that the global web traffic would reach 4.8 zettabytes by the end of 2022[3]. However, the vast amount of information stored in this flood of unstructured data could not be analysed by just applying classic statistical methods. Instead, developing techniques and algorithms capable of discovering and extracting useful patterns from these data are required.

In the context of textual data, this has led to the development of techniques for the automatic analysis of texts, i.e. text mining. Textual data is a basic example of unstructured data because text documents are the most fundamental and natural form of communication through which to store and spread information. Text mining represents an essential step in the KDT process (Feldman & Dagan, 1995), aiming at developing methods and algorithms able to automatically extract useful and important information from texts by analysing a vast amount of words and structures typical of natural language, on the one hand,

[2] https://www.seagate.com/files/www-content/our-story/trends/files/idc-seagate-dataage-whitepaper.pdf

[3] https://twiki.cern.ch/twiki/pub/HEPIX/TechwatchNetwork/HtwNetworkDocuments/white-paper-c11-741490.pdf

and managing the vagueness, uncertainty, and ambiguity that usually characterise texts, on the other hand (Allahyari et al., 2017; Usai, Pironti, Mital, & Mejri, 2018).

Although various kinds of text can be retrieved from different sources of information, this work will focus on a specific type of textual data, the text news, as explained in the following.

1.3 Aim and structure of the thesis

This thesis focused on a particular type of textual data: the news. The possibility of collecting and analysing a large amount of data in "news databases" provides a huge contribution to the study of social trends in time. News media data can be employed to inquire into the transmission and perception of events with respect to which the attention and sensitivity of public opinion have increased.

In the last decades, the need to gather information from large textual datasets has led to the development of automated information extraction methods (Lancichinetti et al., 2015; Usai et al., 2018). Among these methods, those aimed at identifying topics have become very popular in machine learning and natural language processing (Alghamdi & Alfalqi, 2015). However, these methods have shown some limitations, especially when applied to a particular type of text, like the so-called "short texts", such as the news.

For this reason, in recent years, in addition to the development of valuable extensions of the classical topic models, network-based procedures for topic detection in large collections of documents have gained attention in the context of textual analysis, also as an alternative to classical topic models (Hamm & Odrowski, 2021). These methods

are based on the idea that any text can be represented as a word co-occurrence network, where topics emerge as groups of strongly connected words. In addition, the network can be used to explore and present the relations between the topics.

However, although many works have used network-based procedures for detecting topics in textual data, there is a lack of systematic analysis of how different design choices affect the final results in terms of detected topics. Moreover, another unexplored question in the literature about network-based topic detection concerns its relationship with classical topic models, such as the Latent Dirichlet Allocation (LDA; Blei, Ng, & Jordan, 2003) model.

Therefore, the aim of this thesis is two-fold: i) analyse the effect of the relevant design choices on the final results when using network-based procedures to analyse textual data and discover topics, allowing the identification of the fundamental aspects that should be taken into account in this kind of analysis. The goal is to show how and to what extent the choices made during the design phase affect the results, shedding light on which design choices may require further research; ii) compare the network-based approach and the classical topic models on different aspects, such as the coherence of the detected topics, so as to highlight differences and similarities.

More specifically, the present work aims at answering the following research questions:

RQ1 Could the text preprocessing and, consequently, the keyword selection affect the results regarding the features of the discovered topics?

RQ2 Does manipulation in defining the word co-occurrence matrix has an impact on the quality of the discovered topics?

RQ3 Is the detected topics' number and content influenced by the community detection algorithm chosen? If so, to what extent?

RQ4 When analysing text news data, could the network-based procedure for detecting topics represent a valid alternative to classical topic models?

Thus, the main contributions of this work are, on the one hand, the evaluation of the relationship between the shape of the network, which changes depending on the word co-occurrence matrix, the community detection algorithm employed and the features of the discovered topics, and on the other hand, the comparison between network-based procedures and classical topic models for detecting topics, highlighting pros and cons of the network-based approach with respect to the classical topic models when analysing text news data.

The thesis is organised as follows. After the presentation of the fundamental elements of text analysis and classical topic models (Chapter 2), a brief description of the basic aspects of text network analysis and the process of the network-based procedures for topic detection is provided (Chapter 3). Then, Chapter 4 presents the outline and the results of the systematic analysis regarding design choices on a widely used corpus of news articles using the network-based approach. In particular, this chapter explains all the choices made in the text preprocessing step, the definition of the word co-occurrence matrix, and the rationale behind the selection of the community detection algorithms. Moreover, the chapter shows a comparison between the network approach and some of the most famous classical topic models. Finally, the results obtained by applying the network-based approach for topic detection on Italian news articles about distance learning (Chapter 5) are explored. Conclusions end the thesis.

Chapter 2

Textual data analysis

As it has been said in Chapter 1, classical computer analysis methods can not simply process the vast amount of information stored in unstructured data: developing techniques and algorithms capable of discovering and extracting information from these data is required.

In the context of textual data, this has led to the development of techniques for the automatic analysis of texts, i.e. text mining. Text mining is a young and interdisciplinary discipline developed at the intersection of information retrieval, machine learning, statistics and computational linguistics (Allahyari et al., 2017).

Applying text mining methods to large groups of text documents is a very complex operation: the textual data (unstructured) extracted from these documents must be converted into structured data and adequately represented to facilitate their subsequent analysis. For this purpose, text preprocessing and text representation methods are applied. The most important text representation models are the Bag-of-Words (BoW) model and the Vector Space Model (VSM). Then, on the data converted into a suitable format and adequately represented, data mining techniques, such as classification, clustering, and information

extraction methods, are applied (Alghamdi & Alfalqi, 2015; Allahyari et al., 2017).

2.1 Text preprocessing

The application of text preprocessing techniques plays a fundamental role in text mining, mainly because its execution affects the success of the subsequent analysis phases, influence extensively analysed in literature (for example, see Kadhim, 2018; Srividhya & Anitha, 2010; Uysal & Gunal, 2014).

The text preprocessing phase consists of a series of steps. The choice of which steps apply and in which order depends exclusively on the nature and characteristics of the texts analysed and the user's objectives. Despite this, it is useful to focus on those steps that are generally considered the most important for the text preprocessing phase, namely tokenisation, filtering, normalisation, lemmatisation and stemming. In particular, filtering, lemmatisation/stemming and normalisation are used to reduce the size of the dictionary, i.e. the set of words used to describe the document collection (for further information, Allahyari et al., 2017; Kannan & Gurusamy, 2014; Nayak, Kanive, Chandavekar, & Balasubramani, 2016).

2.1.1 Tokenisation

Tokenisation can be defined as the division of text into a sequence of elements called tokens. The token is identified as the smallest part of the text, namely an entity that can not be further subdivided into

smaller parts (Hotho, Nürnberger, & Paaß, 2005; Palmer, 2000).

Tokenisation can be considered as a form of text segmentation. Generally, segmentation is performed by considering alphabetic and alphanumeric characters as tokens delimited by non-alphanumeric characters, such as punctuation marks or whitespace. Thus, in their simplest form, tokens correspond to single words. Moreover, the tokenisation step often requires removing all punctuation marks and replacing all delimiters and non-alphanumeric characters with whitespace, otherwise considered as tokens themselves. The list of tokens is then used in the further text preprocessing phases. The dictionary comprises all tokens obtained by segmenting all text documents in the collection (Allahyari et al., 2017).

It is worth noting that in order to achieve token identification by automatic methods successfully, some issues need to be addressed. Among these, the most important is the one relating to the definition of the token itself because this definition depends on the parsed language and the methods applied (Grefenstette & Tapanainen, 1994; Webster & Kit, 1992). A common problem the researchers encounter during this step regards the definition of token delimiters, i.e. when a non-alphanumeric character can be used or not as a delimiter. In fact, not all non-alphanumeric characters can be interpreted as delimiters. A typical example is represented by the dot, the use of which as a delimiter could be misleading in certain cases, such as when it is part of an abbreviation. However, the development of tokenisation automatic methods allowed us to partially solve this problem (Butler, Wermelinger, Yu, & Sharp, 2011; Weiss, Indurkhya, Zhang, & Damerau, 2005).

Usually, the token is represented by a single word. For this reason, herein, there will be used the words “term” or “word” to refer to the single token.

2.1.2 Filtering

Filtering is applied to texts to remove those words which, depending on the analysed language and the user's goals, are considered not significant for the objectives of the analysis (Allahyari et al., 2017).

For example, a common filtering method consists of removing the so-called "stopwords". Stopwords are words that, although they frequently occur in the texts, have little or no information content, such as prepositions or articles. Similarly, it is believed that even words that occur rarely are irrelevant to the analysis (especially when it concerns a document collection and not an individual text). For this reason, also these words are usually removed. An example of words that occur very rarely is given by typos, which occur especially when the analysed text is written in natural language. Other examples can be provided by names or words written in languages other than that of the document collection, something that frequently happens when downloading texts from the Web. Lastly, an example of words removed because very rarely is given by the "hapaxes", namely the words that occur just once in the documents collection.

For the removal of the stopwords, pre-compiled lists, called "stoplists", are usually used, which differ according to the parsed language. Using stoplists to remove stopwords presents various advantages and disadvantages, bringing more or less relevant benefits based on the type of parsed language. Therefore, in the last years, there have been developed methods for the identification of stopwords, as discussed in Saif, Fernández, He, and Alani (2014).

2.1.3 Normalisation

Normalisation is an important text preprocessing step, as it consists of standardising all the texts by recognising names or other entities of general interest (Bolasco, 2005).

The normalisation step is usually characterised by the following:

- Recognise proper nouns (to prevent them from being confused with common nouns), standardise words that are stressed or separated by hyphens, identify dates or numbers;
- Convert all the words in the lowercase format, a necessary step to avoid splitting effects of the textual data (redundancy);
- Identify compound words (multi-words).

2.1.4 Lemmatisation and stemming

Lemmatisation consists of reducing all the conjugated and inflected forms of a word to a common form so they can be analysed as single items. In other words, lemmatisation methods aim to reduce verb forms to the infinitive tense and nouns to their lemma. The lemma is the form of a word that can be found in the dictionary, so lemmatisation aims to bring all inflected words back to that common form.

However, lemmatisation brings a series of drawbacks (such as the time-consuming or the difficulty in evaluating the results of the process in large texts). Therefore, it is more common to use stemming instead of lemmatisation (Allahyari et al., 2017; Kannan & Gurusamy, 2014).

Regarding stemming, this text preprocessing technique aims to bring all words in a text document back to their root (based on the

hypothesis that inflected words are semantically similar to their root). Stemming does not affect the meaning or the predictive capacity of the model. Still, it allows a significant reduction of the dictionary (for this reason, the occurrence of the words is usually computed on the stemmed terms).

As for the lemmatisation, stemming presents some drawbacks based on the choices made during the analysis and on the parsed language. Despite that, stemming represents one of the most commonly used text preprocessing methods for reducing dictionary size (Hull, 1996).

Stemming can be carried out in two ways:

- In the less radical case, only the word's suffix is deleted, keeping the prefix. This is because it is believed that prefixes, being able to change the meanings of words (for example, from positive to negative), have their inherent information content. In this case, the word obtained's final form is defined as "stem";
- In the more radical case, both the suffix and the prefix are removed (hence, it is a more radical form of stemming). Although it results in a more significant reduction of the dictionary than in the previous case, the loss of information is also more significant. The word obtained's final form is defined as "root" in this second case.

2.1.5 Linguistic text preprocessing

Text mining methods can generally be applied without further text preprocessing steps than those previously described. Sometimes, however, it is necessary to use additional linguistic-type preprocessing

techniques to improve the quality of the words extracted information (Allahyari et al., 2017; Hotho et al., 2005). For this reason, the following linguistic preprocessing methods are frequently used:

- Part-of-Speech (PoS) tagging, which allows the recognition of elements of speech (e.g. nouns, verbs, adjectives) by adding a tag at the end of the words;
- Text chunking, which aims to group adjacent words in a sentence (very useful for identifying compound words);
- Word Sense Disambiguation (WSD), which tries to solve the problems of ambiguity of the meaning of words or sentences, considering the terms not as graphical forms but according to their meaning.

It turned out, however, that for text mining applications, linguistic text preprocessing involves limited improvements compared to the simple application of text representation techniques and the main preprocessing steps, without considering that these techniques are particularly prone to error and require high computational time (Allahyari et al., 2017).

2.2 Text representation

The most common methods for representing text are the Bag-of-Words (BoW) model and the Vector Space Model (VSM).

2.2.1 Bag-of-Words model

The simplest and most common way to represent a text document is to describe it as a set of words, a method known as the Bag-of-Words (BoW) model. This model leads to a text representation that considers the number of occurrences of each term in a document regardless of their order.

In general, to assign a value to a certain term t in document d, based on the importance of the term t in document d, the most straightforward approach is to assign each term t a weight equal to the number of occurrences of the term t in document d. This simple weighting scheme is defined as term frequency ($tf_{t,d}$ or simply tf), where the two subscripts indicate the term and the reference document, respectively (for more information, see Allahyari et al., 2017). For a document d, the set of weights computed with the tf (or with any weight function that determines the number of occurrences of t in d with positive real values) can be seen as a quantitative summary representation of a text. This idea leads to the BoW model.

However, this weighting scheme has a fundamental drawback: all terms are considered equally important within the whole document collection (and not within the single text). On the contrary, some words have greater or lesser discriminatory power and, therefore, more or less importance in the collection. For this reason, it became essential to introduce a mechanism that allows the most relevant terms to be singled out, reducing the importance of those terms that occur so often in the collection that they are no longer significant. The idea behind this mechanism is to scale the weights tf of terms with a high collection frequency, defined as the total number of occurrences of a term in the document collection. The goal is to reduce the weight of a term by

using a value that grows with its collection frequency. However, this is a relatively complex approach, considering that even the smallest document collections have dictionaries consisting of thousands of words.

For this reason, the focus was aroused on the documents rather than the terms, leading to the definition of the inverse document frequency (idf_t or idf), defined as follows:

$$idf_t = log\frac{N}{df_t}, \tag{2.1}$$

where df_t refers to the number of documents in the collection that contain the term t, whereas N is the total number of documents in the collection. Based on this measure, a rare term will get a high weight, while a frequent term will probably get a low weight.

However, the idf measure does not consider that a word frequently occurring in a small set of documents may represent a significant hint of the topics discussed in those documents. The $tf\text{-}idf_{t,d}$ (or $tf\text{-}idf$), a measure that normalises the tf measure by means of the idf measure, gets rid of this problem, assigning high weights to words that occur very frequently in a small set of documents, relatively low weights to words that occur a few times in a document or several times in a wide set of documents, very low weight to words that occur in a huge set of documents (the extreme case is when a word occurs in all the documents, taking a $tf\text{-}idf$ value equal to 0).

The $tf\text{-}idf$ is defined as follows:

$$tf\text{-}idf_{t,d} = tf_{t,d} \cdot idf_t. \tag{2.2}$$

At this point, it may be possible to consider each document as a vector where each component corresponds to each term included in

the dictionary, characterised by a weight which could be the *tf-idf* or any of the measures discussed previously. The idea of representing document collection as a set of vectors is at the base of the Vector Space Model (Schütze, Manning, & Raghavan, 2008).

2.2.2 Vector Space Model

The idea to represent a set of documents as vectors in a common vector space leads to the Vector Space Model (VSM).

In this model, a vector of weights is derived from each document d, denoted as $\vec{V}(d)$, with one component for each term in the dictionary (the words not included in document d take a value equal to 0). Unless otherwise specified, it will always be assumed that the weights are calculated using the *tf-idf* weighting scheme.

Starting from these documents' representation, it is possible to define a similarity measure between two documents in the vector space. A first attempt could be to consider the value of the vector difference between two document vectors. However, this measure has a drawback: two documents with even very similar content can be significantly different simply because one of the two is much longer than the other. Therefore, the relative distribution of terms could be the same in the two documents, but the absolute frequency was not. To compensate for the effect of document length, a standard method for quantifying the similarity between two documents $\vec{V}(d_1)$ and $\vec{V}(d_2)$, is to compute the cosine similarity of the vectors and represent them by using the following expression:

$$sim(d_1, d_2) = \frac{\vec{V}(d_1)\vec{V}(d_2)}{|\vec{V}(d_1)||\vec{V}(d_2)|}, \tag{2.3}$$

where the numerator represents the inner product of the vectors $\vec{V}(d_1)$ and $\vec{V}(d_2)$, while the denominator is given by the product of their Euclidean distance (Manning, Raghavan, & Schutze, 2008).

Moreover, considering a collection of documents as a set of vectors leads to the matrix representation of the collection: it is indeed possible to represent documents and words as the document-term matrix (DTM), a matrix whose rows represent the documents and whose columns correspond to terms; it is also possible to represent documents and words as a term-document matrix (TDM), transposing the DTM (Schütze et al., 2008); finally, it is possible to generate a word co-occurrence matrix (better explained in Chapter 3), a matrix in which each entry is equal to the number of times two words co-occur in the same document (in some cases, within a specific window size).

2.3 Classical topic models

Organising text documents represents one of the main goals for applying data mining methods to document collections. Existing methods for structuring document collections can be generally divided into two categories: methods aimed at assigning keywords to documents based on information relating to the content of the texts; automated information extraction methods (Hotho et al., 2005).

Among the latter, those aimed at identifying topics have become very popular in machine learning and natural language processing (Alghamdi & Alfalqi, 2015), such as probabilistic topic models.

In this work, some of the most popular models employed in the literature for topic discovery will be used, namely the Latent Dirichlet

Allocation model, the Non-negative Matrix Factorization, the Latent Semantic Indexing, BERTopic and BERTopic-MPNET. Specifically, LDA is a generative statistical model, NMF and LSI use a linear algebra approach for topic extraction, while BERTopic and its variant BERTopic-MPNET make use of an embedding approach. The goal is to compare these models with the network-based approach.

2.3.1 Latent Dirichlet Allocation (LDA) model

The Latent Dirichlet Allocation (LDA) model, introduced by Blei et al. (2003), is a probabilistic generative model which assumes that each document in a corpus is generated by a mixture of topics.

More in-depth, the idea at the base of the LDA model is that documents can be represented as random mixtures of latent topics, where each topic is modelled as a probability distribution over the words contained in a fixed vocabulary (Blei et al., 2003). All documents in the collection share the same set of topics (established a priori), which are latent and are to be inferred, but with different proportions. The aim of the LDA model, and topic modelling techniques in general, is to uncover topics from a collection of documents automatically.

It is worth noting that the LDA model is based on the BoW model, according to which the word ordering in the text is negligible for analysis purposes, an assumption called "interchangeability" in the field of probability theory (Blei, 2012). The LDA model has been mainly used and studied in the field of natural language processing, but in the last decade, it has also been used to analyse other kinds of data, such as images (Iwata et al., 2007) and videos (Wang, Sabzmeydani, & Mori, 2007).

The generative process underlying the LDA model can be described generically as follows. For each document in the collection, the words are generated in a two-step process: in the first step, a topic distribution is chosen; in the second step, in order to generate the words of a document, a topic is first drawn from the distribution defined in the first step, and then a word is drawn from the corresponding distribution defined on the vocabulary of that topic. By repeating this process iteratively, it is assumed that the documents are generated. This statistical model reflects the intuition that documents present and consist of a mixture of different topics (for more details, see Blei, 2012; Blei et al., 2003).

Formally, the data-generating process under the LDA model is as follows:

1. Draw topic $\beta_k \sim Dirichlet(\phi)$;
2. For each document d:
   (a) Draw topic proportions $\theta_d \sim Dirichlet(\alpha)$;
3. For each word j:
   (a) Draw topic assignment $z_{d,j} \sim Multinomial(\theta_d)$;
   (b) Draw word $w_{d,j} \sim Multinomial(\beta_{z_{d,j}})$.

The key inferential problem that needs to be solved in order to use LDA is that of computing the posterior distribution of the hidden variables given a document:

$$p(\theta, z|w; \phi, \alpha) = \frac{p(\theta, z, w|\phi, \alpha)}{p(w|\phi, \alpha)}. \tag{2.4}$$

Then, each document is assigned to the topic with the highest posterior probability on that document. For example, suppose to have

estimated the LDA model setting a priori a number of topics equal to 3. Suppose also that the (a posteriori) topic proportions for the i-th document are 0.8 for topic 1, 0.15 for topic 2 and 0.05 for topic 3. As this document deals most with topic 1, the corresponding label is attached. Regarding the interpretation of a topic, this is done by looking into the words given more weight (or probability) under that topic. Note that the only information available is word counts.

The remaining latent variables are inferred via Gibbs sampling (Griffiths & Steyvers, 2004) or Variational Inference (Blei et al., 2003). Herein, it is used the Gibbs sampling as it is more accurate. The real advantage of Variational Inference lies in that it is much faster when the corpus is large. In fact, with Gibbs sampling, it is possible to sample from the true posterior in Equation 2.4, while in Variational Inference, it is chosen the distribution that is closer in a Kullbal-Leibler sense, thus not being more accurate than the true posterior (Blei, Kucukelbir, & McAuliffe, 2017). However, some studies not directly related to the LDA model have shown that despite providing approximate posteriors, variational point estimates can still have good properties (Yao, Vehtari, Simpson, & Gelman, 2018). Extensions like Black-Box VI (Ranganath, Gerrish, & Blei, 2014) show that variational algorithms can provide better predictive densities in relatively short computational times than Gibbs sampling.

2.3.2 Non-negative matrix factorization (NMF)

Non-negative matrix factorization (NMF) is a group of algorithms which has been shown to be a useful decomposition for multivariate data (D. D. Lee & Seung, 1999). More specifically, it is a dimensionality

reduction method that decomposes a matrix into the product of two lower-dimensional matrices, with the property that all elements in the matrices are non-negative (Févotte & Idier, 2011; D. D. Lee & Seung, 1999). Notice that this decomposition is only approximated. Nonetheless, NMF is a popular technique for topic detection in natural language processing, which has been shown several advantages over other topic detection methods, such as LDA (O'Callaghan, Greene, Carthy, & Cunningham, 2015).

In the context of textual analysis, NMF is typically used to detect topics in a document-term matrix, representing the frequency of words in a collection of documents. Let's denote the document-term matrix as V, with dimensions $N \times M$, where N is the number of documents and M is the number of words. Then, NMF aims to decompose V into the product of two matrices, W and H, with dimensions $N \times K$ and $K \times M$, respectively, where K is the number of topics. K is usually chosen such that $NK + KM \ll NM$, hence reducing the data dimension (Févotte & Idier, 2011).

The decomposition is given by the following equation:

$$V \approx WH \tag{2.5}$$

where W and H are non-negative matrices. The elements of W can be interpreted as the weights of the topics in each document, while the elements of H can be interpreted as the weights of the words in each topic.

The first step of applying NMF to the document-term matrix is to initialise W and H matrices with random non-negative values (D. D. Lee & Seung, 1999). Then, NMF aims to find the matrices W and H that best approximate V, subject to the constraint that all elements in the matrices are non-negative (D. D. Lee & Seung, 1999).

This is typically done using an iterative optimisation algorithm, such as the multiplicative update (D. D. Lee & Seung, 2000) or alternating least squares (H. Kim & Park, 2008). The objective function that is minimised during the optimisation process generally is the Frobenius norm of the difference between V and the product of W and H, which is given by the following equation:

$$||V - WH||_F^2 = \sum_i \sum_j [V_{ij} - (WH)_{ij}]^2 \tag{2.6}$$

where $|| \cdot ||_F^2$ is the Frobenius norm, and the subscripts i and j denote the rows and columns of the matrix, respectively.

The multiplicative update algorithm updates the matrices W and H using the following equations:

$$W_{ij} \leftarrow W_{ij} \frac{(VH^T)_{ij}}{(WHH^T)_{ij}} \tag{2.7}$$

$$H_{ij} \leftarrow H_{ij} \frac{(W^T V)_{ij}}{(W^T WH)_{ij}} \tag{2.8}$$

where W^T and H^T are the transposes of W and H, respectively. The alternating least squares algorithm updates the matrices W and H using a similar process but with different equations (for more details, see H. Kim & Park, 2008).

Once the matrices W and H have been optimised, they can be used to identify the main topics in the documents by examining the words with the highest weights in each topic (D. D. Lee & Seung, 1999).

One advantage of NMF over LDA is that the resulting topics are more interpretable (O'Callaghan et al., 2015). In LDA, the topics are represented as a mixture of words, which can be difficult to interpret (Blei et al., 2003). In contrast, the topics in NMF are represented as a weighted sum of words, which makes it easier to understand the main

topics of the documents (D. D. Lee & Seung, 1999). Another advantage of NMF is that it is better suited for handling sparse data (C. J. Lin, 2007). In many NLP applications, the document-term matrix is very sparse, with most of the entries being zero (Aggarwal, 2018). This can be a problem for LDA, which relies on a Dirichlet prior to smoothing the word distributions (Blei et al., 2003). NMF, on the other hand, does not rely on a prior and is therefore able to handle sparse data more effectively (C. J. Lin, 2007). Additionally, it has been shown that NMF works well with short texts (Chen, Zhang, Liu, Ye, & Lin, 2019).

However, NMF presents some limitations. Among them, one of the main limitations of NMF is the need for careful initialisation of matrices W and H (Boutsidis & Gallopoulos, 2008; D. D. Lee & Seung, 2000). The choice of initialisation can significantly affect the resulting topics, and it is important to choose an appropriate initialisation for the data. Moreover, an important aspect of NMF is the choice of the number of topics (to be set in advance, as in the LDA model), which is represented by the parameter K in the decomposition of the matrix V. In general, the value of K should be chosen such that it is sufficiently large to capture the complexity of the data but not so large that it overfits the data (for a discussion about that see, for example, Greene, O'Callaghan, & Cunningham, 2014; Kodinariya & Makwana, 2013).

NMF has been applied to various NLP tasks, including document classification (D. D. Lee & Seung, 1999) and sentiment analysis (Pang & Lee, 2008). It has also been used in other areas, such as image processing (D. D. Lee & Seung, 2000) and gene expression analysis (Brunet, Tamayo, Golub, & Mesirov, 2004).

2.3.3 Latent semantic indexing (LSI)

Latent semantic indexing (LSI), also referred to as Latent Semantic Analysis (LSA), is a technique for topic detection based on the idea that words used in similar contexts tend to have similar meanings and that these meanings can be represented as vectors in a high-dimensional space (Landauer, Foltz, & Laham, 1998). LSI uses singular value decomposition (SVD), a linear algebra technique, to identify the underlying structure in a data matrix (Berry, Dumais, & O'Brien, 1995). In topic detection, LSI can be used to determine the main topics in a collection of documents by analysing the relationships between the words in those documents (Deerwester, Dumais, Furnas, Landauer, & Harshman, 1990).

To implement LSI, the first step is to create a term-document matrix, which is a matrix that represents the frequency of each word in each document. This matrix is typically sparse, meaning that most entries are zero because most words do not appear in most documents (O'Callaghan et al., 2015; Salton & McGill, 1983), as when analysing short texts, such as ones from social networks. The next step is to apply SVD to the term-document matrix, which decomposes the matrix into three matrices: a term matrix, a document matrix, and a diagonal matrix. The term matrix and the document matrix contain the latent (hidden) dimensions of the data, and the diagonal matrix contains the singular values, which represent the importance of each latent dimension (Hofmann, 1999).

Mathematically, the SVD of the term-document matrix can be written as:

$$A = U\Sigma V^T \tag{2.9}$$

where A is the term-document matrix, U is the term matrix, Σ is the diagonal matrix of singular values, and V^T is the transpose of the document matrix (Berry et al., 1995).

Once the matrices have been obtained, LSI selects a subset of the latent dimensions based on the singular values (Hofmann, 1999). This subset is chosen to capture the majority of the variation in the data while still being small enough to be interpretable (Hofmann, 1999). The resulting matrices can then be used to identify the main topics in the documents by examining the words most strongly associated with each latent dimension (Landauer et al., 1998).

LSI has several advantages over other techniques for topic detection. One advantage is that it is able to handle synonymy, which is the phenomenon of different words having the same meaning (Hofmann, 1999). For example, if a document uses the words "dog" and "canine" interchangeably, LSI will treat them as the same word and not create separate topics for each one (Deerwester et al., 1990). LSI can also handle polysemy, which is the phenomenon of a single word having multiple meanings (Hofmann, 1999). For example, if a document uses the term "bass" to refer to a type of fish and a musical instrument, LSI will create separate topics for each meaning (Deerwester et al., 1990). Another advantage of LSI is that it is able to handle large amounts of data efficiently (Hofmann, 1999). SVD is a computationally intensive technique, but it can be implemented using efficient algorithms that scale well with the size of the data (Berry et al., 1995). This makes LSI well suited for large-scale topic detection tasks, such as analysing the contents of a large corpus of documents (Salton & McGill, 1983).

However, there are also some limitations to LSI. One limitation is that it is based on a linear model of the data, which means that it is not able to capture non-linear relationships between words (Hofmann, 1999). This can be problematic if the data contain complex patterns

that cannot be represented linearly (Landauer et al., 1998). Another limitation is that LSI is sensitive to the choice of parameters, such as the number of latent dimensions to use (Hofmann, 1999). Choosing the wrong parameters can lead to poor results, so it is important to tune them carefully (Landauer et al., 1998).

2.3.4 BERTopic

BERTopic is a BERT (Bidirectional Encoder Representations from Transformers; Devlin, Chang, Lee, & Toutanova, 2018) based topic modelling technique (Grootendorst, 2022). BERT is a deep learning model that has been trained on a large dataset of text and has achieved state-of-the-art performance on a variety of natural language processing tasks, including language understanding, language generation, and machine translation (Devlin et al., 2018).

BERT is based on transformer architecture, a type of neural network that can process sequential data in a parallel manner. The transformer architecture uses self-attention mechanisms to capture the dependencies between input sequences and has been shown to be effective at modelling long-range dependencies in texts (Devlin et al., 2018).

BERT and its variations (e.g., Lan et al., 2019; J. Lee et al., 2020; Liu et al., 2019), such as BERTopic and BERTopic-MPNET (Grootendorst, 2022; Song, Tan, Qin, Lu, & Liu, 2020), have shown remarkable results in generating contextual word and sentence vector representations. The semantic properties of these vector representations allow the meaning of texts to be encoded so that similar texts are close in vector space (Grootendorst, 2022).

BERT is used as an embedder, while BERTopic provides document

embedding extraction, making use of a sentence-transformer (Sentence-BERT or SBERT; Grootendorst, 2022) framework in order to create a vector space in which to compare embed document representations semantically using pre-trained language models. This is done in order to cluster semantically similar documents. Then, for reducing the dimensionality of document embeddings generated, BERTopic makes use of UMAP (Uniform Manifold Approximation and Projection) algorithm (McInnes, Healy, & Melville, 2018) to handle high-dimensionality data and HDBSCAN, a hierarchical density-based clustering model (Campello, Moulavi, & Sander, 2013; Grootendorst, 2022). However, users can use whatever clustering methods they prefer.

Topics are generated from these clusters using a class-based *tf-idf* algorithm in which words are grouped into classes based on their semantic meaning (Egger & Yu, 2022; Grootendorst, 2022). The importance of a word is then measured within its class rather than across all documents. This allows BERTopic to capture the meaning of words better and to identify topics that are relevant to a specific domain. (Sánchez-Franco & Rey-Moreno, 2022). This means that the higher the value is for a term, the more representative it is of its topic.

About BERTopic-MPNET, it is a variant of BERTopic which uses a different embedding model, namely the "all-mpnet-base-v2", whereas BERTopic uses by default the "all-MiniLM-L6-v2" model, which works well with English documents.

The main difference between BERTopic and classical topic models, such as LDA, is that the former provides continuous rather than discrete topic modelling (Alcoforado et al., 2022), thus leading to different results with repeated modelling due to the stochastic nature of the model. Once the model is computed, researchers can output the most important topics (Grootendorst, 2022).

One advantage of BERTopic is that, such as NMF, it works well

with short texts, identifying, however, more clear-cut topics than NMF (Egger & Yu, 2022; Grootendorst, 2022). For a more detailed analysis of the outperformance of BERTopic over other topic models, such as LDA, see Egger and Yu (2022) and Zihan, Meng, Ling, and Mohammad-Reza (2022).

2.3.5 Drawbacks of topic models

Despite their popularity and numerous applications in different fields, such as sociology, history and linguistics, topic models are known to suffer from conceptual and practical problems.

One example is given by the difficulty of properly choosing the number of topics that has to be defined a priori. In fact, choosing the wrong number of topics can lead to poor performance of the model: if the number of topics is too large, the model may overfit the data and produce poor results; on the other hand, if the number of topics is too small, the model may underfit the data and produce poor results (Griffiths & Steyvers, 2004). Determining the optimal number of topics for a given dataset can be challenging without extensive experimentation (Steyvers & Griffiths, 2007; Wallach, Murray, Salakhutdinov, & Mimno, 2009). The determination of the number of topics has led to a broad debate, during which various orientations have emerged (Buntine, 2009; Chang & Blei, 2009; Wallach et al., 2009).

Moreover, probabilistic topic models can be sensitive to the size of the dataset (Steyvers & Griffiths, 2007). Indeed, these models, in general, perform better with larger datasets, as they can capture the relationships between words and topics more accurately. However, larger datasets also require more computational resources to process, which

can be a limitation for some applications (Steyvers & Griffiths, 2007). For example, the LDA model can be computationally expensive to fit large datasets, as it requires sampling from the posterior distribution of the model using Markov Chain Monte Carlo (MCMC) methods (Blei et al., 2003). Furthermore, the computational complexity of the LDA model scales linearly with the size of the dataset, making it less practical for vast datasets (Griffiths & Steyvers, 2004).

Another drawback of probabilistic topic models is that they can be sensitive to the quality of the input data (Steyvers & Griffiths, 2007). For example, if the input documents contain a lot of noise or irrelevant information, the model may produce poor results. Preprocessing the input data to remove noise and irrelevant information can help improve the performance of the model (Steyvers & Griffiths, 2007), extending, on the other hand, the time needed to carry out the analysis.

Regarding BERTopic, it carries out a document-level analysis, assuming that each document only contains a single topic. Although documents can be divided into smaller parts, such as sentences and paragraphs, it is not an ideal representation (Grootendorst, 2022). Secondly, also BERTopic has shown poor performance when applied to vast amounts of data, making the model as susceptible to the size of the dataset as LDA (de Groot, Aliannejadi, & Haas, 2022).

Nevertheless, topic models are still widely used in natural language processing due to their ability to identify the main topics in a collection of documents (Steyvers & Griffiths, 2007). However, it is important to be aware of these limitations and to carefully consider e.g. the number of topics, dataset sizes and input data quality when using these models.

Chapter 3

Text network analysis and network-based procedures for topic detection

Recently, in the context of textual analysis, network-based procedures for topic detection have gained attention, also as an alternative to classical topic models (Hamm & Odrowski, 2021).

These methods are based on the idea that any text can be represented as a word co-occurrence network, where topics emerge as groups of strongly connected words. Moreover, networks can be used to explore and present the relations between the topics.

In this chapter, a description of the basic elements of text network analysis will be provided, as well as a description of the community detection algorithms employed. Finally, a brief review of some applications of network-based procedures for topic detection in the literature ends the chapter.

3.1 Basics of network analysis

Network analysis is a set of techniques used to depict relations among elements that are somehow connected and analyse the structures that emerge from the recurrence of these relations.

More formally, a network can be described by a graph $\mathcal{G}$ consisting of a pair (V, E), where V is the set of nodes (or vertices) and $E \in V \times V$ is the set of links (or edges) between the nodes. Depending on the network, the graph may be undirected, directed, weighted or unweighted (Wasserman & Faust, 1994).

A directed graph (or digraph) is a graph with edges having orientations. Therefore, in this kind of graph, E is the set of edges, which are ordered pairs of distinct nodes. Let (v_i, v_j) be the relationship between node i and node j. In a directed graph, $(v_i, v_j) \neq (v_j, v_i)$. On the contrary, an undirected graph is a graph in which edges do not have orientations. Then, in this case, $(v_i, v_j) = (v_j, v_i)$.

Regarding weighted and unweighted networks, the first ones are characterised by the fact that each edge between two nodes has an associated weight representing the strength or intensity of the relationship between the nodes. In a weighted network, the weights can be continuous or discrete values, and they can be positive or negative (Newman, 2018). Weighted networks help represent phenomena where the strength of the relationships between nodes is important, such as in social networks (Wasserman & Faust, 1994) or transportation networks (J. Lin & Ban, 2013), for example. On the other hand, in an unweighted network, there are no weights associated with the edges. All edges are treated as having equal strength or intensity (Cohen & Havlin, 2010).

Besides, the relational data represented by a network can be organised in matrix form. In particular, graphs can be described as adjacency or affiliation matrices, which can be used to carry out one-mode and two-mode network analyses, respectively. More specifically, an adjacency matrix is a square matrix used to describe a finite graph. The matrix entries express whether the pairs of a finite set of nodes are adjacent in the graph or not. Instead, the affiliation matrix shows the relationship between two classes of objects, for which the matrix has one row for each element of the first class and one column for each component of the second one (Wasserman & Faust, 1994).

Moreover, in network analysis applications, crucial is the identification of important nodes in the network and detection communities. Regarding the definition of the importance of the nodes, in literature, it is possible to find several measures that represent different aspects of the importance of a node. Among them, the most popular ones are:

- **Degree centrality.** Node degree centrality measures the number of connections a node has in a network (Wasserman & Faust, 1994). A node with a high degree is connected to many other nodes and may play a central role in the network. Node degree is often used to identify nodes central to the overall network structure, and it is a simple measure that can be easily computed for both weighted and unweighted networks (Newman, 2018).

- **Betweenness centrality.** Betweenness centrality measures the number of shortest paths between pairs of nodes that pass through a given node (Freeman, 1977). A node with a high betweenness centrality is a "bottleneck" in the network, as it lies on many shortest paths. Betweenness centrality is often used to identify nodes central to the overall flow of information or

resources in the network (Newman, 2018).

- **Eigenvector centrality.** Eigenvector centrality is a measure of the influence of a node in a network based on the idea that a node's influence is proportional to the influence of its neighbours (Newman, 2006). A node with a high eigenvector centrality is connected to other influential nodes and is considered an influential node. Eigenvector centrality is often used to identify nodes that are central to the overall structure of the network (Newman, 2018).

- **Closeness centrality.** Closeness centrality measures the distance between a node and all other nodes in the network (Sabidussi, 1966). A node with a high closeness centrality is close to many other nodes in the network, and it may play a central role in the overall structure of the network. Closeness centrality is often used to identify nodes central to the overall flow of information or resources in the network (Newman, 2018).

Degree, betweenness, eigenvector and closeness centrality are all measures commonly used in network analysis to identify important nodes in a network (Newman, 2018). These measures are based on different characteristics of the network structure and can be used to identify different aspects of important nodes. Indeed, as stated in Valente, Coronges, Lakon, and Costenbader (2008), the level of correlation among these measures seems nearly optimal because it is neither too high nor too low, indicating that they measure different things. In other words, the amount of correlation between degree, betweenness, closeness and eigenvector centrality measures suggests that these measures are distinct yet conceptually related.

As it will be explained in Chapter 4, in order to determine the importance of each node inside the community, all of these measures will be used to compute the overall coherence of each community.

3.2 Representing text as a network

At a basic level, text network analysis can be regarded as a set of techniques aimed at studying the relationships between words in a text or collection of documents (Schnegg & Bernard, 1996). These techniques are based on the idea that texts can be represented as networks, where the words are the nodes, and the links are given by the semantic relationships between them. By analysing the structure of text networks, it is possible to gain insights into the meaning and organisation of the texts (De Nooy, Mrvar, & Batagelj, 2018).

In text network analysis, word co-occurrence matrices are commonly used, which involve counting the number of times two words appear in the same document (Borgatti, Everett, & Johnson, 2018) or sliding window (Bullinaria & Levy, 2007).

Text network analysis can be applied to a wide range of text types, including natural language texts, such as news articles or social media posts, and structured texts, such as bibliographic data or metadata (Bernard & Ryan, 1998).

Text network analysis can also be used to study the evolution of language and meaning over time (Kleinberg, Kumar, Raghavan, Rajagopalan, & Tomkins, 1999). For example, by analysing the co-occurrence patterns of words in a large corpus of texts, it is possible to study how the meanings of words change over time or how new words emerge and spread through a language (Kleinberg et al., 1999).

This type of analysis can provide insights into the social and cultural context in which the texts were produced, as well as the underlying structures and patterns of language itself (De Nooy et al., 2018).

3.2.1 The reasons behind

Representing texts as networks allows the possibility of capturing the complex relationships between the elements in a text. Though seemingly simple, representing the text using a word co-occurrence matrix, where two words are connected if they appear as neighbours in the text, is valuable and reasonable because it can capture the author's style (Mehri, Darooneh, & Shariati, 2012), textual complexity (Amancio, Aluisio, Oliveira, & Costa, 2012) and many others textual aspects (Masucci, Kalampokis, Eguíluz, & Hernández-García, 2011). For example, Amancio et al. (2012) investigated the changes in writing style in books published over several centuries by making use of metrics for complex networks, finding that just using these metrics with a basic text preprocessing can lead to an appropriate clustering of books that matched the traditional literary classification.

Moreover, network analysis allows for handling large amounts of data (Ding, Chowdhury, & Foo, 2001). Traditional approaches to textual analysis may struggle to scale to large datasets due to the computational complexity of the algorithms involved. On the other hand, network analysis is well-suited for handling large datasets, as it allows for efficient representation and analysis of the relationships between elements. This makes it a valuable tool for tasks such as topic modelling, where the goal is to identify the main topics in a large collection of documents.

In addition to these advantages, network analysis also offers the ability to incorporate additional information into the analysis. For example, network analysis can be used to incorporate the context in which words are used, such as the position of a word in a sentence or the presence of other words in the same document (Tang, Qin, & Liu, 2015). This can be especially useful for tasks such as sentiment analysis, where the meaning of words can be heavily influenced by the context in which they are used.

Overall, network analysis is a powerful tool for textual analysis that offers several advantages, showing to be effective in a wide range of natural language processing tasks (Hofmann, 2001).

3.2.2 Word co-occurrence matrix

The word co-occurrence matrix represents one of the essential elements in text network analysis. It represents the co-occurrence of words in a matrix form, with each row and column representing a unique word and the cells indicating the number of times two words appear together in a text, part of a text or a sliding window around a specific word.

Word co-occurrence matrices provide a concise representation of the relationships between words in a text since they preserve the semantic relationship between them. In addition, representing the co-occurrence of words in a matrix form allows easily visualising the relationships between words and identifying patterns in the data. This can be particularly useful for analysing large texts, as it allows quickly identifying trends and patterns that may not be immediately obvious when reading the text (de Arruda, Costa, & Amancio, 2015; Pennacchiotti & Pantel, 2009).

In this vein, very interesting is the work of Bullinaria and Levy (2007), in which the authors investigated the use of word co-occurrence matrices for extracting semantic representations in large text corpora.

There are different methods for building word co-occurrence matrices. Herein, the sliding window approach will be used, which involves moving a window of fixed size across the text and counting the number of co-occurrences within the window. There are different methods for positioning the window, but authors usually pose it on the right of the target word, so here it was decided to do the same.

3.2.3 Community detection algorithms

Community detection algorithms are a class of algorithms used to uncover the community structure within a network (Lancichinetti & Fortunato, 2009). These algorithms are widely used in many fields, including social network analysis, biological network analysis and natural language processing (see, for example, Devi & Poovammal, 2016; Tripathi, Parthasarathy, Sinha, Raman, & Ravindran, 2019). These algorithms can be divided into two main categories: overlapping and non-overlapping community detection algorithms.

Non-overlapping community detection algorithms are designed to identify a network partition into non-overlapping communities, where each node belongs to only one community. On the contrary, overlapping community detection algorithms are designed to identify communities that overlap, meaning that a node can belong to multiple communities simultaneously (Negara & Andryani, 2018).

One of the most commonly used non-overlapping community detection algorithms is the Louvain algorithm (Blondel, Guillaume, Lambiotte, & Lefebvre, 2008). This algorithm is based on modularity maximisation, but it uses a heuristic approach to identify the network partition that maximises the modularity measure. The Louvain algorithm is a fast and computationally effective algorithm. Thus, it is well-suited for identifying communities in large networks (Blondel et al., 2008; Negara & Andryani, 2018). However, many other community detection algorithms have been proposed in the literature, such as Newman's leading eigenvector algorithm (Newman, 2006).

As regards the overlapping community detection algorithm, the Speaker-Listener Label Propagation Algorithm (SLPA; Xie, Szymanski, & Liu, 2011) has shown to be very effective among the others.

Non-overlapping community detection algorithms

The Louvain algorithm is one of the most popular community detection algorithms used to identify community structure within a network. It is based on the concept of modularity maximisation (Newman, 2006), which is a measure of the density of edges within communities compared to the density of edges between communities. The Louvain algorithm uses a heuristic approach to identify the partition of a network that maximises the modularity measure.

The modularity of a partition is defined as

$$Q = \frac{1}{2m} \sum_{i,j} \left[A_{ij} - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j), \tag{3.1}$$

where A_{ij} is the adjacency matrix of the network, k_i is the degree of node i, m is the total number of edges in the network, $\delta(c_i, c_j)$ is the Kronecker delta function (equal to 1 if $c_i = c_j$ and 0 otherwise) and c_i and c_j are the communities to which nodes i and j belong,

respectively.

The Louvain algorithm follows a two-step process. Firstly, taking a weighted network as input, the Louvain algorithm starts by assigning each node to a community. Hence, in this first step, there are as many communities as there are nodes. Then, the algorithm repeatedly moves nodes between communities to maximise the modularity measure. This process is repeated iteratively for all the nodes until the modularity measure can no longer be improved.

The change ΔQ of modularity obtained by removing node i from its community ad moving it into the community c_j is given by

$$\Delta Q = \left[\frac{\sum_{\text{in}} + 2k_{i,\text{in}}}{2m} - \left(\frac{\sum_{\text{tot}} + k_i}{2m}\right)^2\right] - \left[\frac{\sum_{\text{in}}}{2m} - \left(\frac{\sum_{\text{tot}}}{2m}\right)^2 - \left(\frac{k_i}{2m}\right)^2\right], \tag{3.2}$$

where $\sum_{\text{in}}$ is the sum of all the weights of the edges inside the community c_j, $\sum_{\text{tot}}$ represents the sum of all the weights of the edges incident to the nodes inside community c_j, $k_{i,\text{in}}$ is the sum of the weights of the edges between node i and other nodes in the community c_j, k_i is the degree of node i and m is the total number of edges in the network.

In the second step, the algorithm builds a network in which the communities found in the first step are the nodes and the edges between the nodes inside the communities give the edges between the communities. The edges between nodes inside the same community lead to self-loops for the community in the network.

Once this second step is completed, the algorithm re-apply the first step of this process on this new network, going on until there are no more changes in the modularity, which has reached its maximum.

The Louvain algorithm has been shown to be fast and effective at

identifying the structure of communities in large networks (Blondel et al., 2008).

Newman's leading eigenvector algorithm (Newman, 2006) is a community detection algorithm also based on the concept of modularity maximisation. Like the Louvain algorithm, the leading eigenvector algorithm seeks to identify a network partition that maximises the modularity measure. However, Newman's algorithm tries to achieve this goal by finding the leading eigenvector of the modularity matrix and using this eigenvector to partition the network into communities.

Define the index vector s_i such that

$$s_i = \begin{cases} +1 & \text{if vertex } i \text{ belongs to group 1} \\ -1 & \text{if vertex } i \text{ belongs to group 2,} \end{cases} \tag{3.3}$$

and

$$\frac{1}{2}(1 - s_i s_j) = \begin{cases} 1 & \text{if } i \text{ and } j \text{ are in different groups} \\ 0 & \text{if } i \text{ and } j \text{ are in the same group.} \end{cases} \tag{3.4}$$

Then, it is possible to rewrite Equation 3.1 as

$$Q = \frac{1}{4m} \sum_{i,j} \left[A_{ij} - \frac{k_i k_j}{2m} \right] s_i s_j, \tag{3.5}$$

or, equivalently, in the following matrix form

$$Q = \frac{1}{4m} \mathbf{s}^T \mathbf{B} \mathbf{s} \tag{3.6}$$

which is called the modularity matrix. In this expression, $\mathbf{s}$ refers to the column vector whose elements are s_i, while $\mathbf{B}$ is a real symmetric matrix with elements $B_{ij} = A_{ij} - k_i k_j / 2m$.

Afterwards, rewriting $\mathbf{s}$ as linear combination of the normalised

eigenvector u_i of the matrix $\mathbf{B}$ so as $\mathbf{s} = \sum_{i=1}^{n} a_i \mathbf{u}_i$ with $a_i = \mathbf{u}_i^T \cdot \mathbf{s}$, it is possible to express Equation 3.6 as

$$\begin{aligned} Q &= \frac{1}{4m} \sum_i a_i \mathbf{u}_i^T \mathbf{B} \sum_j a_i \mathbf{u}_i \\ &= \frac{1}{4m} \sum_{i=1}^{n} (\mathbf{u}_i^T \cdot \mathbf{s})^2 \beta_i. \end{aligned} \tag{3.7}$$

In Equation 3.7, β_i is the eigenvector of $\mathbf{B}$ corresponding to $\mathbf{u}_i$. Assuming that the eigenvalues are labelled in decreasing order, the goal here is to find an appropriate division of the network which maximise the modularity, an objective that can be achieved by choosing the index vector $\mathbf{s}$ proportional to the eigenvector $\mathbf{u}_1$. This means concentrating all the weight in the term involving the largest eigenvalue β_1, being the other terms equal to zero because the eigenvectors are orthogonal.

The algorithm described so far allows for dividing the network into two communities: to obtain a division into a larger number of parts the standard approach is to repeat the division of the communities obtained into two parts, then divide those parts and so forth.

As for the Louvain algorithm, when there are no more changes which can increase the modularity, then the algorithm stops.

Therefore, the key difference between the Louvain algorithm and the leading eigenvector algorithm is their approach to identifying the structure of communities in a network. The Louvain algorithm uses a heuristic approach, while the leading eigenvector algorithm uses the leading eigenvector of the modularity matrix (J. Lin & Ban, 2013).

However, both algorithms have shown good results in their time efficiency for large-size networks (Y. Lee, Lee, Seong, Stanescu, & Hwang, 2020) and in their results identifying communities with high modularity scores compared with other methods (Mothe, Mkhitaryan,

& Haroutunian, 2017).

Overlapping community detection algorithms

Overlapping community detection algorithms are used to identify the structure of communities within a network where nodes can belong to multiple communities simultaneously. One such algorithm is the Speaker-Listener Label Propagation Algorithm (SLPA; Xie et al., 2011), a label propagation algorithm (Raghavan, Albert, & Kumara, 2007) designed to identify overlapping network communities.

As explained in Xie et al. (2011), the SLPA algorithm works by iteratively updating the nodes' labels based on their neighbours' labels. At each iteration, each node selects the label most commonly held by its neighbours and adopts it as its own. This process is repeated until the labels of the nodes converge. The final labels of the nodes correspond to the overlapping communities they belong to.

The SLPA algorithm has several parameters that can be adjusted to control its behaviour. For example, when selecting its current label, the "memory" parameter determines the number of previous labels each node remembers. In addition, the "threshold" parameter determines the minimum fraction of neighbours holding a particular label for a node to adopt that label.

SLPA algorithm was tested in both synthetic and real-world networks, showing very interesting results, especially in comparison with other well-known overlapping community detection algorithms (Xie, Kelley, & Szymanski, 2013; Xie et al., 2011).

Measures for evaluating communities

Interpreting communities as topics makes it reasonable to use topic coherence measures to evaluate overall communities' interpretability.

Topics found by topic models or community detection algorithms need to be understandable. Otherwise, the blind application of these algorithms could lead to misleading and meaningless topics. For this reason, topics assessment is complemented by qualitative human evaluations, done by e.g. reading the most important words of each topic or providing a comprehensive assessment of the topic's composition. However, finding human evaluators with prior knowledge about the datasets' field and sometimes assessing hundreds of topics may be difficult and time-consuming.

Topic coherence measures aim to solve these problems giving researchers an easy-to-understand measure able to represent the quality of human perception of the detected topics (Stevens, Kegelmeyer, Andrzejewski, & Buttler, 2012). More specifically, topic coherence measures provide a value of the degree of semantic similarity between high-scoring words in a topic, representing in this way that these words well support a topic (Röder, Both, & Hinneburg, 2015; Rosner, Hinneburg, Röder, Nettling, & Both, 2014).

In literature (Douven & Meijs, 2007, Röder et al., 2015), there are three measures usually used to evaluate topic coherence:

$$C_{UMass} = \frac{2}{N \cdot (N-1)} \sum_{i=2}^{N} \sum_{j=1}^{i-1} \log \frac{P(w_i, w_j) + \epsilon}{P(w_j)},$$

$$C_{UCI} = \frac{2}{N \cdot (N-1)} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \log \frac{P(w_i, w_j) + \epsilon}{P(w_i) \cdot P(w_j)},$$

$$C_{NPMI} = \frac{2}{N \cdot (N-1)} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \frac{\log \frac{P(w_i, w_j) + \epsilon}{P(w_i) \cdot P(w_j)}}{-\log(P(w_i, w_j) + \epsilon)},$$

where $P(w_i)$, $P(w_j)$ and $(P(w_i, w_j)$ represent respectively the ratio between the number of documents containing word w_i, word w_j, and both words w_i, w_j and the total number of documents in the corpus,

while ϵ is a small constant added to prevent logarithm of zero.

In this work, all these measures were used to measure community coherence, comparing the results with the ones obtained using classical topic measures.

3.3 Network-based procedures for topic detection

Essentially, a network-based topic discovery process takes the following form:

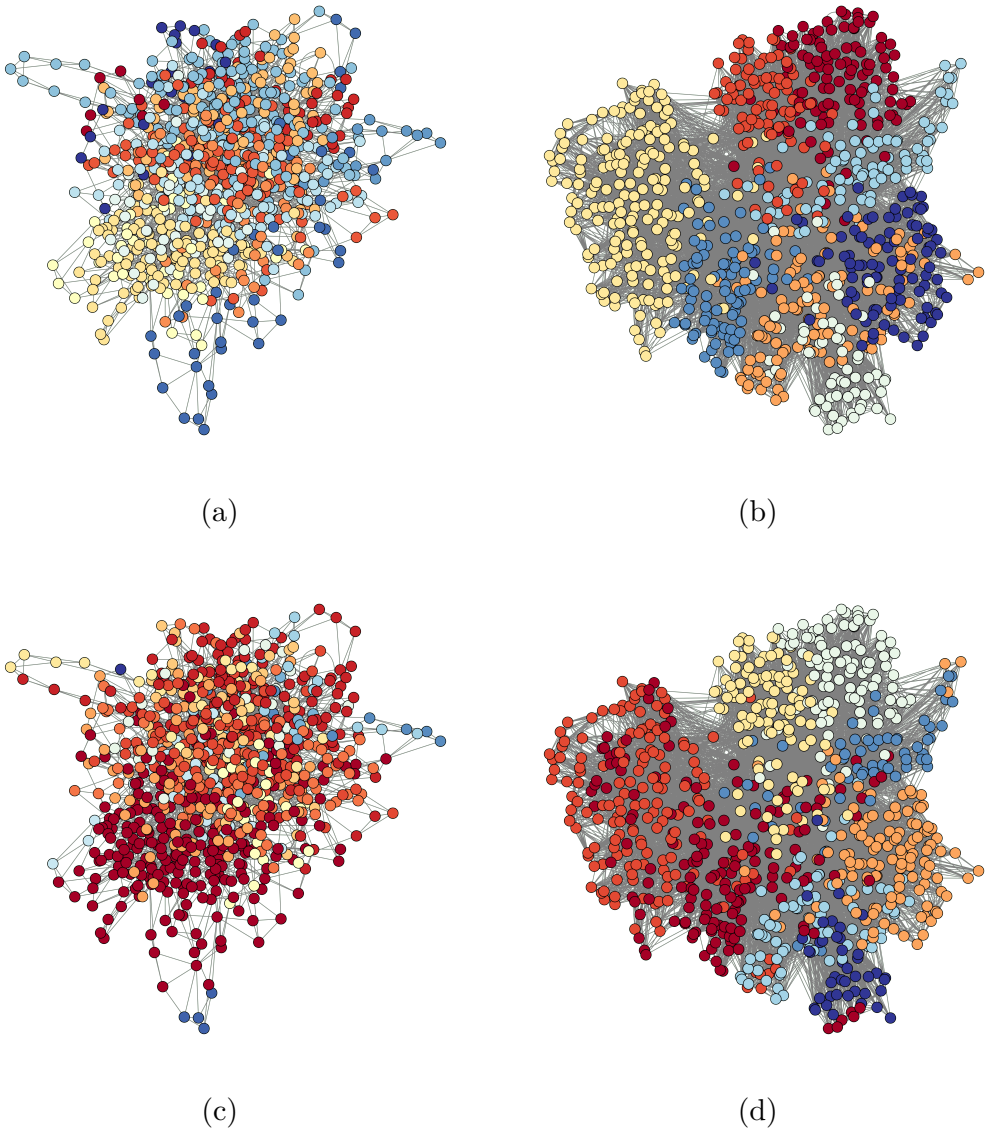
- preprocessing the text, a step-by-step procedure during which the researcher selects which methods to apply to clean the text and make it ready for the analysis (e.g. removal of non-alphanumeric characters, removal of stopwords, reduction of terms to a common root);
- forming of the word co-occurrence matrix by defining the context in which two words will be considered semantically related. This is usually done by defining what is meant by "co-occurrence" between words;
- building of the network and selection of the community detection algorithm. This procedure requires the researcher to make decisions in each of these steps.

This procedure requires the researcher to make decisions in each of these steps.

Although many works have used network-based procedures for detecting topics in textual data, there is a lack of systematic analysis of how different design choices affect the final results in terms of detected topics.

In this thesis, the focus is on the two defining steps of this process, as they are unique to network-based approaches: building the word co-occurrence matrix and selecting the community detection algorithm. This choice is based on the idea that the definition of the word co-occurrence matrix, which determines the shape of the network, and the community detection algorithm employed are strongly related to the characteristics of the discovered topics. Moreover, the impact of other design choices on text classification has already been studied in a non-network context. For instance, Uysal and Gunal (2014) have investigated the impact of text preprocessing on text classification, revealing that choosing an appropriate combination of preprocessing steps may improve classification accuracy.

As an example, Figure 3.1 shows four different networks built using the same documents. They represent the word co-occurrence matrices of 9 news extracted from the BBC news articles collection (Greene & Cunningham, 2006) concerning business, sport, and tech. More specifically, in the first (Figure 3.1a) and the third (Figure 3.1c) networks two words belonging to the same document are adjacent or co-occur if they are at most 2 words apart (that is if, between the two words, there is at most one word in between). On the other hand, the second (Figure 3.1b) and the fourth (Figure 3.1d) networks have been built considering that two words in the same document co-occur if they are at most 10 words apart. Furthermore, in order to identify the topics, the Louvain community detection algorithm (Blondel et al., 2008) was applied on the first and the second networks (Figure 3.1a and Figure



(a) (b)

(c) (d)

Figure 3.1. Example of networks obtained from 9 news of the BBC news article collection. In networks (a) and (c) the window size is equal to 2, while it is equal to 10 in networks (b) and (d). The colours represent the community to which each node belongs according to community detection algorithms: the Louvain algorithm in (a) and (b) and Newman's leading eigenvector method in (c) and (d). Note that the organisation of nodes in communities varies between networks. Indeed, while in (b) and (d) the organisation in communities is clear, in (a) and (c) the partition is much less defined.

3.1b), while on the other two networks was applied Newman's leading eigenvector method for detecting communities (Newman, 2006). It is possible to observe how the shape of the networks and the detected communities change. For example, it is possible to observe more defined communities in the networks with a window size equal to 10, some communities recognised by one method are split into two by the other, and some nodes are assigned to a different community.

Analysing the effect of the relevant design choices on the final results allows to identify the fundamental aspects that should be taken into account when using network-based procedures to analyse textual data and discover topics and those which may require further research. Therefore, the main contribution of this work is to evaluate the relationship between the shape of the network, which changes depending on the word co-occurrence matrix, the community detection algorithm employed, and the features of the discovered topics.

Another unexplored question about network-based topic detection is about its relationship with probabilistic topic models, such as the LDA model. While this question is also important, before addressing it is necessary to develop a deeper understanding of optimal design choices for network-based methods. Therefore, in Chapter 4, after exploring the relationship between the choices made in the network-based approach and the discovered topics, the comparison between this approach and the classical topic models will be examined.

3.3.1 Applications of network-based procedures for topic detection in the literature

In recent years, many works have been written about applying community detection methods for topic discovery.

For example, Sayyadi and Raschid (2013) find topics as communities in a keyword co-occurrence matrix using the Girvan-Newman community detection algorithm based on the betweenness centrality measure. They build the keyword occurrence matrix considering that two keywords are connected if they co-occur in at least one document, and the weight of that link is given by the number of documents in which both keywords co-occur. Then, they compute each word's document frequency and remove the links with a value below a specific threshold.

Another example is given by Salerno, Tataru, and Mallory (2015), who apply the Louvain community detection algorithm for discovering topics on a weighted network in which nodes represent individual words in the vocabulary and links indicate the co-occurrence of a pair of words within a document. The weight of the links between words is determined by the context in which two words co-occur: for example, a co-occurrence within the same sentence carries more weight than a co-occurrence within the same paragraph. Then, they evaluate their results using modularity and compare the error rate to the results achieved by two baselines: one that classifies documents randomly and another one that classifies documents based on the most common label in the training set. Similar approaches can be found in Dang and Nguyen (2018).

Instead, de Arruda et al. (2015) investigate how specific definitions of the occurrence between words favour the emergence of communities

of semantically related words, allowing for the identification of relevant topics. In particular, they consider three different ways to define the co-occurrence between two words in the pre-processed text: two words are connected if they are separated by at most a given number of other words; words belonging to the same paragraph are linked together in a clique, disregarding links between words further from each other than the given maximum distance; finally, the statistical significance of co-occurrences with regard to random, shuffled texts is tested. The fast-greedy method is used to find communities of high modularity.

Lancichinetti et al. (2015) discover topics using the Infomap algorithm on networks built, considering that two words are connected if they co-occur in the same document. More specifically, they compute the dot product similarity of each pair of words that co-occur in at least one document in order to compare it against the expectation for a null model where words are randomly shuffled across documents. Then, a threshold is defined for retaining words for which the co-occurrence between them cannot be explained by the null model. However, because Informap is run as a non-overlapping community detection algorithm, to cope with generic words used in multiple topics, they refine the results obtained from applying the community detection algorithm using a latent topic model that allows for non-exclusivity.

Some of the most recent contributions in this area are given by M. Kim and Sayama (2020) and Hamm and Odrowski (2021). The former transforms the textual data into a vector form by computing the tf-idf (term frequency-inverse document frequency) score considering each sentence as a document. Afterwards, they compute the pair-wise cosine similarity of the tf-idf vectors to build adjacency matrices of the sentences. Then they use the Louvain community detection algorithm on the sentence networks, where the nodes are the sentences, and

the cosine similarity of tf-idf representations between every node pair represents the link weight.

Hamm and Odrowski (2021) apply the Leiden community detection algorithm on undirected weighted networks investigating the effects of the resolution parameter on modularity maximisation. Moreover, they define a measure to identify the most significant words within a topic.

This thesis contributes to this research line by considering the relationship between the definition of the word co-occurrence matrix, the selection of the community detection algorithms, and the final results.

Chapter 4

Systematic analysis

4.1 Goals and motivations

As stated in Chapters 1 and 3, in the context of textual analysis, network-based procedures for topic detection are gaining attention, also as an alternative to classical topic models. Network-based procedures are based on the idea that documents can be represented as word co-occurrence networks, where topics are defined as groups of strongly connected words.

Although many works have used network-based procedures for topic detection, there is a lack of systematic analysis of how different design choices, such as the building of the word co-occurrence matrix and the selection of the community detection algorithm, affect the final results in terms of detected topics. Another unexplored question about network-based topic detection concerns its relationship with classical topic models, such as the LDA model.

Therefore, the aim of this chapter is to address these questions by developing a deeper understanding of optimal design choices for network-based procedures, showing how and to what extent the choices

made during the design phase affect the results, and comparing network-based procedures for topic detection with classical topic models in terms of overall discovered topics' interpretability.

More specifically, this work focuses on a particular type of textual data: the news. Therefore, the main analyses presented in this chapter were carried out on the BBC news article collection. Then, to assess the validity of the investigated approach, all the analyses were carried out on two other well-known news collections, namely the 20 NewsGroup dataset and the Reuters-21578 dataset. Finally, a comparison with topic models ends the chapter.

4.2 Research outline

In this section, the data and the tested design choices are described.

4.2.1 BBC dataset

The analysis focuses on the corpus of BBC news articles, a collection of documents widely used as a benchmark for machine learning research (Greene & Cunningham, 2006). The collection comprises 2,225 complete news articles (2,090 after removing the duplicates) collected from 2004 to 2005 and divided into five topics: business, entertainment, politics, sport and technology.

The total number of articles and unique words per topic is reported in Table 4.1. In the analysis, both the headline and the body of each news were considered.

Table 4.1. Number of documents and unique words for each topic of the BBC news articles collection.

Topic	Documents	Unique words
Business	500	10,790
Entertainment	366	11,040
Politics	395	10,636
Sport	492	9,997
Technology	337	11,444

4.2.2 Data preprocessing

For all the experimental conditions, in the preprocessing stage non-alphanumeric characters, numbers and words composed of 1 or 2 characters were removed. Afterwards, the text was divided into tokens, choosing the single word as the unit of analysis. Then, the stopwords were removed using a list provided with the dataset, and in order to reduce the vocabulary size, that is, the set of unique words used in the text corpus, the text was stemmed (Allahyari et al., 2017). Also, the *hapaxes* (words with a frequency equal to 1) were removed.

At this point, two different text cleaning approaches were carried out: in the first one, all the preprocessed words obtained in the text preprocessing described above were retained; in the second one, a PoS tagger was employed, keeping at the end of the preprocessing step only nouns, adjectives and adverbs.

Finally, to remove very common words not included in the stopword list, in some experimental conditions words with a value of the *tf-idf* less than 0.01, 0.1 and 1 were filtered out.

4.2.3 Word co-occurrence matrix

Once preprocessed the corpus and obtained the vocabulary, the word co-occurrence matrices were built. To generate the word co-occurrence matrices, the number of times two words co-occur in the same document within a specific window size was counted.

There are three ways of positioning the window around the target word: to the left of the word, to the right or on either side (Bullinaria & Levy, 2007). Herein, windows of different sizes placed to the right of the words were considered, as usually done in the literature. In this work, window sizes equal to 2, 5, 10, 15 and 20 were used.

Furthermore, in the literature, many authors apply different filters to the word co-occurrence matrix based on the distribution of the words or their frequency in order to reduce the size of the matrix. For this reason, this aspect was tested by using different filters for the word co-occurrence matrices. More specifically, the 100, 500 and 1000 words with the lowest co-occurrence values and the 50, 100 and 500 words with the highest co-occurrence value were removed.

Afterwards, inspired by Salerno et al. (2015), who applied different weights based on the context in which two words co-occur, experimental conditions were defined by modifying the co-occurrence values assigned to words within the window size. In particular, weights proportional to the words' proximity were assigned. For example, for a window size equal to 3, the word adjacent to the target word gets a value equal to 1; the next word takes a value equal to 2/3; then, a value equal to 1/3 was assigned to the last word.

Table 4.2 shows, as an example, the first group of experimental conditions carried out. The other six groups of experiments (exp2.*, exp3.* and so on) differ in the condition defined in the last column,

namely in the filter applied on the words frequency distribution described above. As regards the other columns of the table, the first column reports the labels of the experimental conditions, the column labelled as "Keyword extraction" indicates if the text was applied a normal or a PoS keyword extraction, while the column "Tf-idf filter" refers to the *tf-idf* value used to get rid of common words not included in the stoplist. The column "Weighting scheme" refers to the weighting scheme used for counting words co-occurrence. Finally, the last two columns show the size of the windows and the filter applied to the word frequency distribution.

In total, 112 experimental conditions per window size were carried out.

4.2.4 Network analysis and community detection algorithms

Starting from the word co-occurrence matrices, interpreted as weighted adjacency matrices, the undirected weighted networks were built on which three different community detection algorithms were applied.

Since almost all the works reported in the literature review (Section 3.3.1) applied modularity optimisation algorithms, it was decided to use the Louvain community detection algorithm as one of the most popular among them; nevertheless, in previous analyses, also the Leiden algorithm was tested, which is claimed to outperform the Louvain algorithm, which can sometimes lead to bad-connected communities (Traag, Waltman, & Van Eck, 2019). However, with respect to the scope of this work, the results of the Leiden algorithm were very similar to those obtained using the Louvain algorithm. For this reason, only

Table 4.2. Description of the first group of experimental conditions. The other groups of experimental conditions differ in the condition defined in the last column.

Experiment	Keyword extraction	Tf-idf filter	Weighting scheme	Window size	Matrix filter
exp 1.1	normal	0.01	weighted	2, 5, 10, 15, 20	none
exp 1.2	PoS	0.01	weighted	2, 5, 10, 15, 20	none
exp 1.3	normal	0.01	proximity	2, 5, 10, 15, 20	none
exp 1.4	PoS	0.01	proximity	2, 5, 10, 15, 20	none
exp 1.5	normal	0.1	weighted	2, 5, 10, 15, 20	none
exp 1.6	PoS	0.1	weighted	2, 5, 10, 15, 20	none
exp 1.7	normal	0.1	proximity	2, 5, 10, 15, 20	none
exp 1.8	PoS	0.1	proximity	2, 5, 10, 15, 20	none
exp 1.9	normal	1	weighted	2, 5, 10, 15, 20	none
exp 1.10	PoS	1	weighted	2, 5, 10, 15, 20	none
exp 1.11	normal	1	proximity	2, 5, 10, 15, 20	none
exp 1.12	PoS	1	proximity	2, 5, 10, 15, 20	none
exp 1.13	normal	0	weighted	2, 5, 10, 15, 20	none
exp 1.14	PoS	0	weighted	2, 5, 10, 15, 20	none
exp 1.15	normal	0	proximity	2, 5, 10, 15, 20	none
exp 1.16	PoS	0	proximity	2, 5, 10, 15, 20	none

the results obtained from the Louvain algorithm were presented in this work. Then, to investigate the performance of a different kind of approach, a spectral algorithm was employed, namely Newman's leading eigenvector method. The rationale behind this choice is that if the network obtained after the preprocessing phase presents clearly separated topics, different algorithms should find similar results. In contrast, for networks with a less clear community structure, the specific types of community each different method is designed to identify would potentially lead to significantly different results.

Finally, herein it argues that despite the absence of the application of methods finding overlapping communities in the literature on network-based topic detection, these methods should be the most appropriate in theory. In general, it could not exclude that a word belongs to multiple topics at the same time, but using a partitioning method (as the two algorithms mentioned above) prevents the identification of such cases. Therefore, the hypothesis stated here is that overlapping community detection algorithms should perform better than non-overlapping community detection algorithms, as they allow words to belong to different communities simultaneously. As a consequence, the SLPA algorithm was tested as a method designed to discover overlapping communities (Xie et al., 2011). Note that as an overlapping community detection algorithm, the K-clique algorithm (Palla, Derényi, Farkas, & Vicsek, 2005) with different values for the k parameter was also tried. Still, it did not manage to obtain results probably because of the presence of large dense subgraphs, making this approach computationally intractable.

4.2.5 Inside the communities

Looking into hundreds or thousands of randomly ordered words constituting each community is certainly not helpful for grasping its meaning. Instead, it would be necessary to rank the words somehow in order to look at only the most characteristic ones.

In the context of network analysis, it is possible to recognise several measures aimed at defining the importance of a node. However, each of these measures grabs only one aspect of the importance of a node: for example, the node degree centrality focuses on the number of connections a node has in a network; on the other hand, betweenness centrality is a measure of the number of shortest paths between pairs of nodes that pass through a given node.

To test which of the measures presented in Section 3.1 may be considered more suitable for identifying the top n-words within each community, all of them were used to rank the words inside the communities, evaluating the results by applying the topic coherence measures (Section 3.2.3).

4.2.6 Back to the text

Starting from the assignment of the words to the communities, it is also possible to determine which document of the collection is concerned with which topics.

In the literature about network-based topic detection, there are no clear clues about how to achieve this goal. However, this problem is discussed in Hamm and Odrowski (2021); there, the solution was to count the relative number of topic terms within a document. In fact,

they stated that although it may seem a simple approach, especially when compared with e.g. probabilistic topic models, the count of topic terms works very well when one uses a network-based approach.

For this reason, in this work, the same approach was followed. Then, in order to evaluate the results, precision, recall and F_1-score measures were computed.

Defining as "true positives" (TP) the documents correctly classified by the model (meaning that they are included in the topic they belong to in the original dataset), as "false negatives" (FN) the documents clustered in topics different from the actual topic they belong to and as "false positives" (FP) the documents assigned to a topic but belonging to another one, it is possible to define the precision as the ratio between the true positive and the sum of true positives and false positives, and the recall as the ratio between the true positives and the sum of true positives and false negatives. The F_1-score represents a metric that combines recall and precision using the harmonic mean.

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{4.1}$$

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4.2}$$

$$F_1\text{-score} = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \tag{4.3}$$

4.3 Results and discussion

This section presents the results of the experiments, focusing on how the different choices made in the preprocessing stage, the definition

of the word co-occurrence matrix and the selection of the community detection algorithm affect the features of the detected topics.

4.3.1 The effect of the window size and the text preprocessing

The main result observed is that the number of communities obtained by the algorithms is generally higher for smaller window sizes. Indeed, as the window size increases, the number of communities the algorithms find decreases, remaining constant for window sizes greater than 5.

Figures 4.1, 4.2 and 4.3 show the number of communities found applying the three algorithms on the word co-occurrence matrices for all the experimental conditions: here, the number of communities identified by the non-overlapping community detection algorithms, that is, the Louvain and leading eigenvector methods (Figures 4.1 and 4.2), is always greater than the number of communities identified by SLPA for window sizes greater than 2 (Figure 4.3). In particular, SLPA finds only one community with these settings.

It should be noted that to check the robustness of the results obtained from the tested algorithms, which are not deterministic, all the experimental conditions were run 30 times.

Interestingly, the choices made during the text preprocessing stage seem not to affect the results in terms of the number of communities found. In fact, looking at the figures, it is possible to observe that the number of communities found by the three algorithms remains stable in the different experimental conditions, even if the vocabulary size is smaller when cleaning the text using a PoS tagger.

To give an example, in the first run, the number of unique preprocessed words in experiment 1.1 was equal to 9,913, while in experiment 1.2 the number of unique preprocessed words was equal to 8,078.

4.3.2 Filters on the word co-occurrence matrix

The results remain stable when removing the words with the lowest co-occurrence values from the word co-occurrence matrices.

Instead, removing the words with the highest co-occurrence values changes the number of detected communities for almost all the windows size tested: for example, when removing the top 500 words for window sizes equal to 2, the Louvain community detection algorithm finds from 5 to 31 communities (Figure 4.2), Newman's algorithm finds between 2 and 5 communities (Figure 4.1), while the SLPA finds until 12 communities (Figure 4.3). The results for window sizes greater than 5 become stable, although, in some cases, far away from the collection's actual number of topics. For instance, when removing the top 500 words for window sizes equal to 20, the Louvain algorithm finds about 10 communities (Figure 4.2).

Interestingly, even if it seems that Newman's algorithm finds the correct number of communities in the experimental conditions 7.*[1] for window sizes equal to 2 or 5, in fact, the communities are not balanced, with often three big communities and two residual ones composed of 10 words at most. The same happens when applying the SLPA algorithm, where there is usually only one big community, with the

[1]Note that "7.*" means "experimental condition 7.1", "experimental condition 7.2" and so on, instead, "*.1", for example, means "experimental condition 1.1", "experimental condition 2.1" and so on.

Figure 4.1. Number of communities found by Newman's eigenvector algorithm per window sizes for all the experimental conditions on the BBC dataset. On the y-axis is the number of communities found, while on the x-axis are the experimental conditions. For each window size and experimental condition, the algorithm was run 30 times, with each line representing a run. Here, "WS" means "window size".

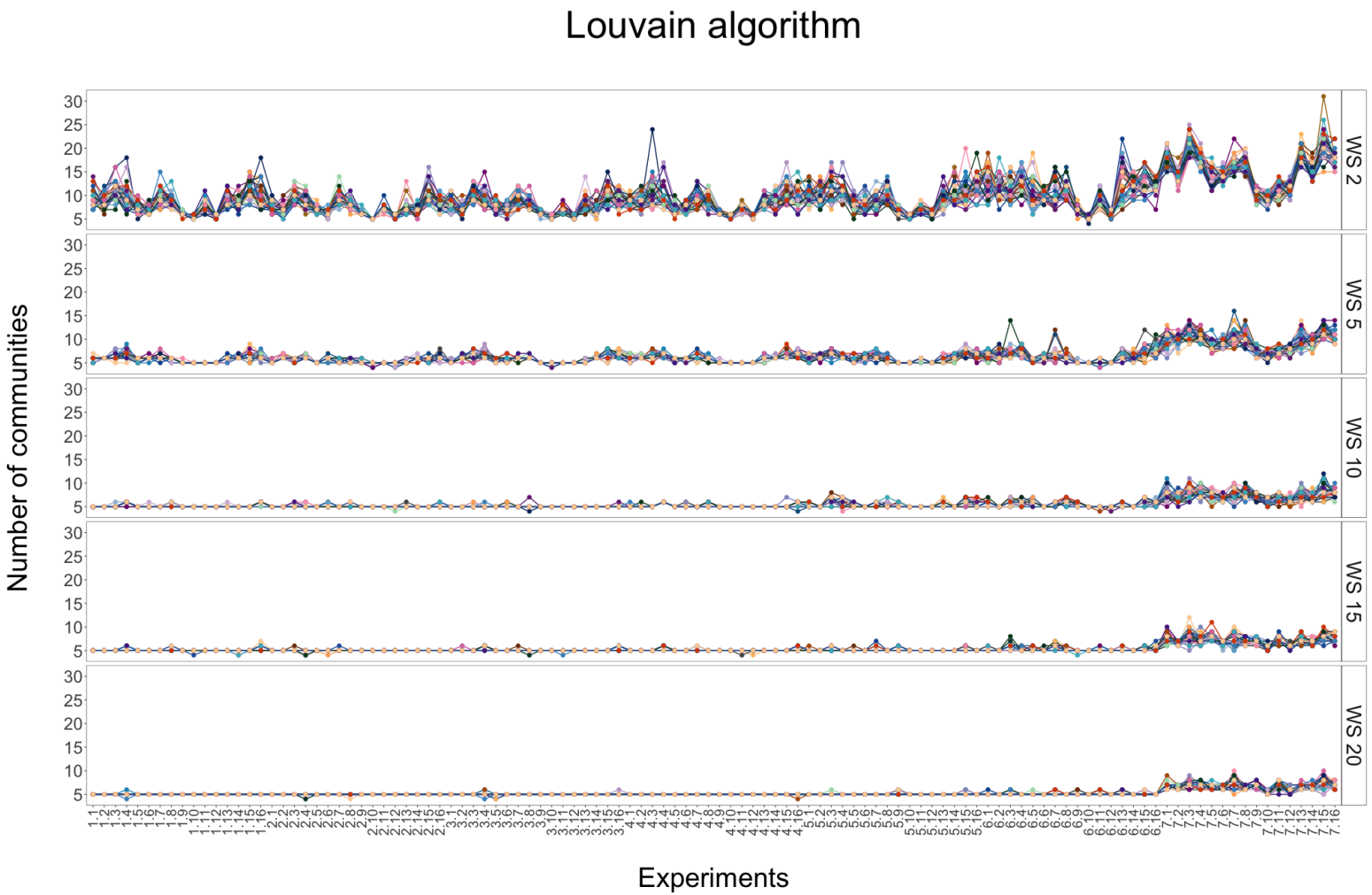


Figure 4.2. Number of communities found by the Louvain algorithm per window sizes for all the experimental conditions on the BBC dataset. On the y-axis is the number of communities found, while on the x-axis are the experimental conditions. For each window size and experimental condition, the algorithm was run 30 times, with each line representing a run. Here, "WS" means "window size".

Figure 4.3. Number of communities found by the SLPA algorithm per window sizes for all the experimental conditions on the BBC dataset. On the y-axis is the number of communities found, while on the x-axis are the experimental conditions. For each window size and experimental condition, the algorithm was run 30 times, with each line representing a run. Here, "WS" means "window size".

others consisting of just tens or hundreds of words.

4.3.3 Weighting scheme

The effect of using a different weighting scheme within the window sizes was assessed. Looking at Figures 4.1, 4.2 and 4.3 it is possible to infer that the use of a different weighting scheme does not affect the results in terms of the number of communities found, both considering different window sizes and the community detection algorithms employed.

4.3.4 Selection of the community detection algorithm

Finally, regarding the community detection algorithm, the Louvain algorithm shows the most interesting results. In almost all the experimental conditions, this algorithm finds a number of communities equal to the number of the actual topics in the document collection for window sizes greater than 5. Moreover, as shown in Figures 4.4a, 4.4b and 4.4c, the communities are coherent with the content of the actual topics in the BBC collection, with each community representing mainly one topic.

Note that Figure 4.4 was built by matching the communities' words of experiments 1.1 for the Louvain algorithm and 7.1 for the SLPA algorithm of the first run with the actual topics' words, enabling possible overlapping[2]. Therefore, in the representation of the correspondence

[2]Note that these experimental conditions were chosen not for particular reasons,

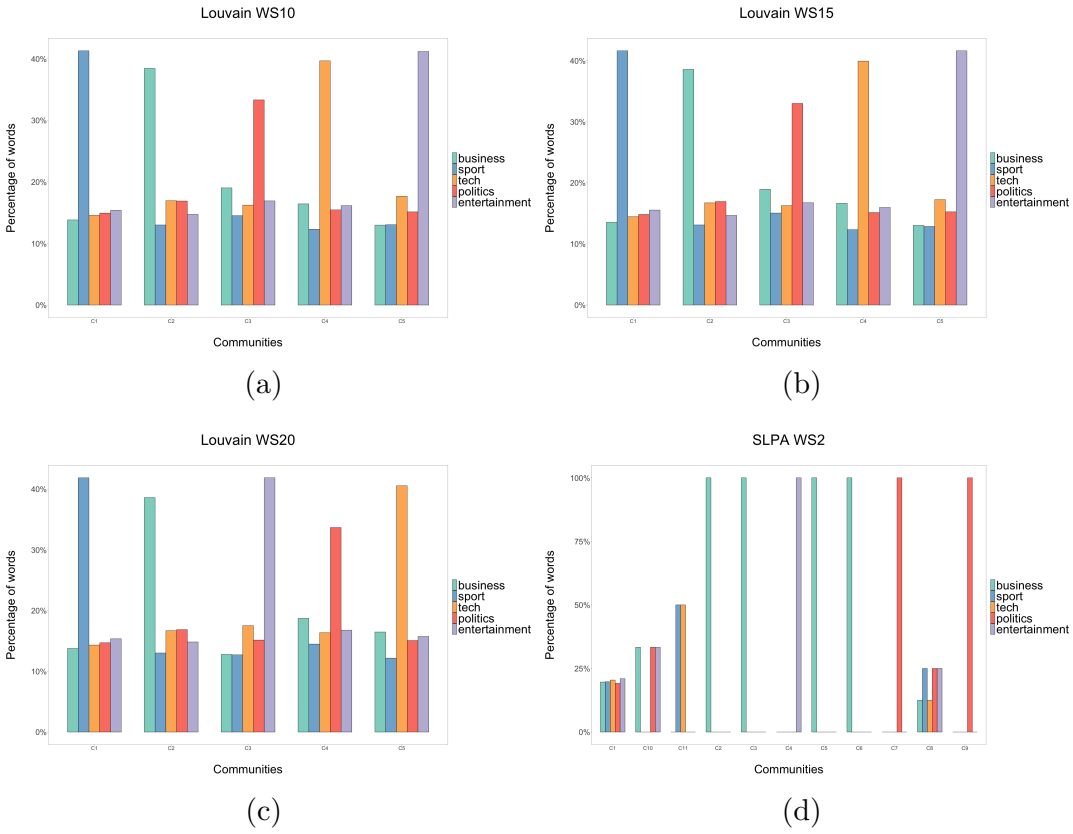


Figure 4.4. Matching of the communities' words to actual topics' words for the Louvain community detection algorithm for window sizes equal to 10 (a), 15 (b) and 20 (c), and for the SLPA algorithm for window size equal to 2 (d). On the y-axis is reported the percentage of words in each community belonging to each actual topic (every group of bars sum up to 100%), while on the x-axis are the communities. Here, "C" means "community", while "WS" means "window size".

between communities' words and topics' words, generic words such as "month" could be included in more than one topic.

To better understand these results, the communities found by the Louvain community detection algorithm for a window size equal to

but to give an example.

10 (Figure 4.4a) in experimental condition 1.1 could be an example. First, the size of communities is quite balanced, with the number of words ranging from 2,202 to 2,955. Then, from an inspection of the words with the highest node degree within each community (Table 4.3), it is possible to observe that they are coherent with the topic they represent. So, for example, among the top 15 words with the highest node degree in the last community (community "C5"), there are words such as "show", "film", "star" and "music", coherent with the topic "entertainment".

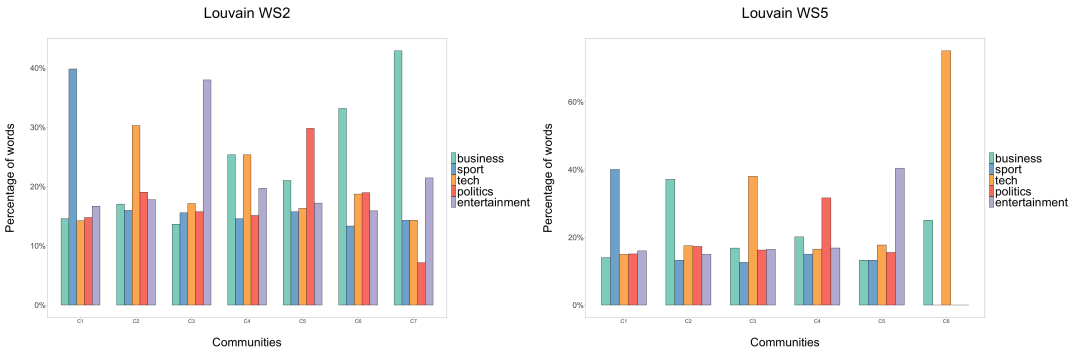
The same can be observed for window sizes equal to 15 and 20. Instead, in the cases in which the Louvain algorithm finds more than 5 communities, namely for window sizes equal to 2 and 5, it could be observed that there are always 5 bigger communities coherent with the original topics and a variable number of smaller communities (as shown in Figure 4.5). Moreover, the largest communities generally include a number of words greater than 2,000, whereas the smallest are composed of hundreds, tens, or just a few words.

To provide a more detailed analysis of the communities identified by the Louvain algorithm under different settings, the Adjusted Rand Index (ARI; Hubert & Arabie, 1985), a metric for comparing disjoint clustering solutions, was computed on the results of experiment 1.1 of the first run. Table 4.4 shows the ARI for different window sizes. It is possible to observe that the ARI is generally high, particularly between the partitions obtained considering window sizes greater than 5. More specifically, for window sizes greater than 5, ARI values range from 0.666 to 0.871, showing high similarities, but also that the algorithm finds the same number of communities but the communities are not identical.

The lowest ARI values are associated with the partitions obtained using smaller window sizes, requiring additional analysis to show

Table 4.3. Top 15 words with the highest node degree centrality measure under each community found in the experimental condition 1.1 in the BBC dataset for window size equals to 10. In parenthesis, the degree centrality score is rounded to two decimals.

Community 1	Community 2	Community 3	Community 4	Community 5
game (0.49)	year (0.67)	govern (0.55)	peopl (0.47)	film (0.51)
first (0.48)	compani (0.52)	say (0.46)	technolog (0.46)	includ (0.47)
play (0.47)	market (0.50)	told (0.45)	work (0.44)	star (0.47)
time (0.47)	firm (0.47)	minist (0.45)	get (0.43)	best (0.42)
win (0.46)	month (0.42)	labour (0.42)	way (0.40)	award (0.42)
two (0.44)	report (0.41)	parti (0.42)	servic (0.39)	show (0.39)
back (0.42)	countri (0.41)	plan (0.41)	comput (0.39)	music (0.38)
against (0.41)	expect (0.41)	claim (0.40)	user (0.38)	perform (0.36)
second (0.40)	share (0.40)	elect (0.39)	call (0.37)	top (0.29)
before (0.39)	sale (0.3)	public (0.38)	phone (0.36)	director (0.28)
player (0.38)	price (0.38)	right (0.37)	help (0.36)	british (0.27)
just (0.38)	busi (0.38)	issu (0.37)	want (0.36)	number (0.27)
world (0.37)	product (0.37)	law (0.37)	system (0.35)	actor (0.27)
final (0.36)	analyst (0.36)	blair (0.36)	mobil (0.35)	name (0.25)
three (0.36)	group (0.36)	tori (0.36)	network (0.35)	follow (0.25)



(a) (b)

Figure 4.5. Matching of the communities' words to actual topics' words for the Louvain community detection algorithm for window sizes equal to 2 (a) and 5 (b). On the y-axis is reported the percentage of words in each community belonging to each actual topic (every group of bars sum up to 100%), while on the x-axis are the communities. Here, "C" means "community", while "WS" means "window size".

how these communities relate to those found using larger window sizes. Moreover, the contingency table between the partitions obtained with window sizes equal to 5 and 10, respectively, was computed to understand better the tendency of the algorithm to merge communities related to the same topic by increasing the window size (Table 4.5). The table shows that some of (but not all) the clusters obtained using a window size equal to 5 are assimilated into some larger clusters found in the partition obtained using a window size equal to 10.

The two other algorithms fail to find a reasonable number of communities, with the SLPA algorithm finding only one community for window sizes greater than 2 in all the experiments. Even in those cases where SLPA finds more than one community, the communities are not balanced, with almost all the words within one of the detected communities. Figure 4.4d shows the results obtained by applying the

Table 4.4. Value of the ARI computed between all the partitions obtained by the Louvain community detection algorithm applied to networks built from the different word co-occurrence matrices. Here, "WS" means "window size".

	WS2	WS5	WS10	WS15	WS20
WS2	1				
WS5	0.419	1			
WS10	0.391	0.717	1		
WS15	0.376	0.698	0.838	1	
WS20	0.370	0.666	0.822	0.871	1

Table 4.5. Contingency table between Louvain community detection algorithm partitions obtained considering window sizes equal to 5 and 10. Here, "C" means "Community", while "WS" means "window size".

Louvain WS5 Louvain WS10	C1	C2	C3	C4	C5	C6
C1	2,658	50	37	122	88	0
C2	69	1,881	61	147	44	0
C3	91	68	93	1,937	73	0
C4	74	70	1,894	65	68	3
C5	124	32	65	68	2,319	0

SLPA algorithm on the word co-occurrence matrix with a filter on the top 500 words (experiment 7.1) using a window size equal to 2. Note that in the first community, there are 11,668 words, while in the others the number of words ranges from 1 to 3.

4.3.5 Evaluation of the discovered topics

In order to assess the quality of the discovered topics, measures for evaluating topic coherence were applied. Moreover, assignment performance indicators were computed.

Because of the similar results obtained in each run by applying the three algorithms (Figures 4.1, 4.2 and 4.3), the results of the first run were employed in the remainder of this section to continue the analysis. Moreover, considering the performance of the Louvain algorithm, in the remainder of this chapter only its results will be shown.

Topic coherence

As described in Section 3.2.3, topic coherence measures provide a value of the degree of semantic similarity between high-scoring words in a topic, representing in this way that these words well support a topic.

Network analysis offers different measures for defining the importance of a node that could be used to rank words inside the communities, making it possible to apply topic coherence measures for evaluating the overall interpretability of a topic.

Herein, the most commonly used network centrality measures, namely degree, betweenness, eigenvector and closeness centrality measures, were employed to rank the words inside the communities and compute the UMass, UCI and NPMI topic coherence measures. The results of applying these measures for all the experimental conditions using the Louvain algorithm with a window size equal to 20 are shown in Figure 4.6. The choice of using the results obtained using a window size equal to 20 lies in the robustness of the results (see Figure 4.2).

Looking at Figure 4.6, it is possible to observe that the results obtained from the four network centrality measures are almost identical.

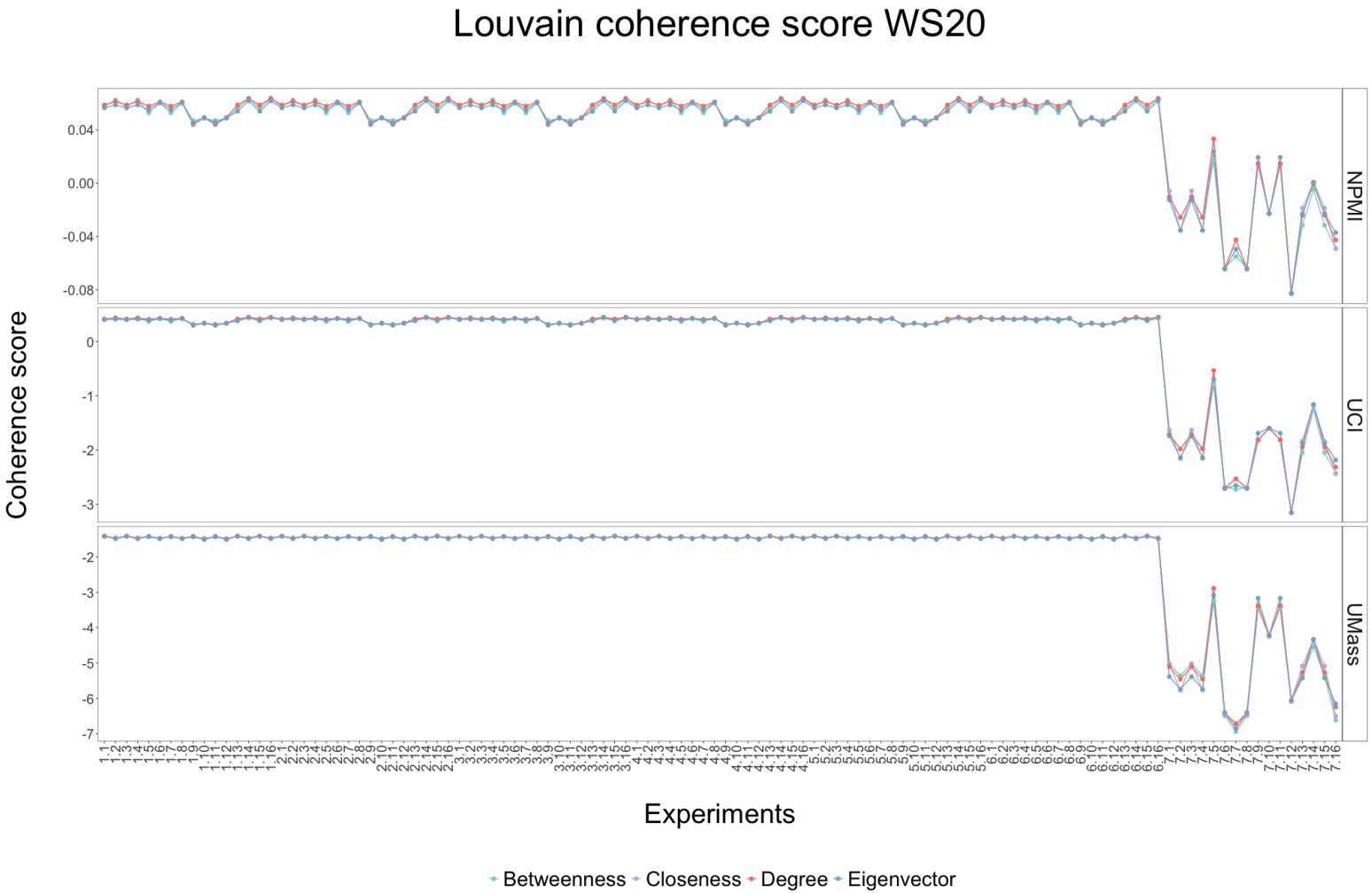


Figure 4.6. Topic coherence scores for all the experimental conditions using the Louvain community detection algorithm for a window size equal to 20 on the BBC dataset. On the y-axis are reported the topic coherence scores, while on the x-axis are the experimental conditions. Note that the y-axis has different scales for each coherence method.

An explanation could lie in the presence of dense subgraphs in the network, which makes the choice of which measures to use for ranking the words meaningless because the measures give the same results. For this reason, it was decided to use the results obtained using the degree centrality for comparing the topic coherence scores of the network-based approach with the classical topic models' scores. Notice that, in general, a higher topic coherence score implies a better topic's interpretability. Moreover, it should also be noted that topic coherence measures are affected by the number of detected topics: the greater the number of topics, the lower the topic coherence score.

In this vein, the first result that emerges is the unsuitableness of the filter on the top 500 words, for which the coherence scores are the lowest. Then, more interesting are the results obtained using both UCI and NMPI coherence measures (Figure 4.6). Indeed, it is possible to observe a pattern showing lower coherence values for experiments from *.9 to *.12, which has not emerged from the inspection of the number of topics found. Among the others, these experimental conditions are characterised by removing words with a *tf-idf* lower than 0.1. This result, concurrently both with the coherence scores obtained for the experiments group 7.* and the observation that for the first six groups of experiments the number of communities found by the Louvain algorithm is the same, shows that the removal of common words should be done with rationale, considering the balance between the need to reduce the vocabulary size and, at the same time, to preserve the sentences semantic flow.

Another interesting result that emerges from Figure 4.6 is the pattern in the scores for normal and PoS text cleaning, with the latter obtaining better results in all the experimental conditions (regardless of the experiments group 7.*).

Experiment 1.13 is the one with the highest UMass score, while

experiment 1.14 is the one with the highest UCI and NPMI coherence scores. Interestingly, experimental conditions 1.13 and 1.14 are characterised by the use of the weighted weighting scheme for the word co-occurrence matrix definition and no filter based on the *tf-idf* value. The two experimental conditions differ only in the text cleaning approach employed (normal or PoS).

However, experimental condition 1.1 was chosen for computing assignment performance indicators. The rationale behind this choice is that, first of all, although the (slightly) difference in the coherence scores between the first experimental condition (UMass = -1.410104, UCI = 0.4190436 and NPMI = 0.05857704) and experimental conditions 1.13 (UMass = -1.406222) and 1.14 (UCI = 0.4536163 and NPMI = 0.06369224), the former is characterised by the most common text preprocessing and vocabulary size reduction choices employed in literature; secondly, the choice of not applying a filter based on the *tf-idf* value adopted in experimental condition 1.13 and 1.14 does not take into account that when using a non-overlapping community detection algorithm, such as the Louvain algorithm, it is possible that the algorithm could correctly cluster together words belonging to a single topic while arbitrarily including multi-topic words in only one of the communities where they should have been included. This could represent a problem when computing precision, recall and F_1-score measures in a network-based context, in which, as it has been stated in Section 4.2.6, the approach is to count the relative number of topic terms within a document.

Precision, recall and F_1-score

Thus, picking up experiment 1.1, precision, recall and F_1-score measures were computed to evaluate the clustering obtained using the network-based approach. The results are shown in Table 4.6.

The results are very good, in line with the ones obtained in literature (Hamm & Odrowski, 2021). In section 4.5 will be shown the comparison with the ones obtained by applying classical topic models.

Table 4.6. BBC classification statistics for predicting preassigned classes by detected topics from the network-based approach in the experimental condition 1.1.

Topic	Precision	Recall	F_1-score
Business	0.91	0.92	0.92
Entertainment	0.97	0.88	0.92
Politics	0.93	0.99	0.96
Sport	0.97	0.83	0.90
Technology	0.84	0.93	0.88
Average	0.92	0.91	0.92

4.4 Evaluation on others news articles collections

To verify the effectiveness of the network-based approach on news textual data, all the experimental conditions presented in the previous section were carried out on two other real-world text collections: the

20 Newsgroups dataset and the Reuters-21578 dataset.

On these datasets, the same data preprocessing applied to the BBC news article collection was carried out, except that for the hapaxes removal: in fact, in these two datasets, the words with a frequency equal to or lower than 10 (and not equal to 1) were filtered out. This had to be done due to the presence of several typos and spurious words with low frequency.

4.4.1 20 Newsgroups (20NG) dataset

The 20 Newsgroups (20NG) dataset is a widely used dataset for text mining applications. Collected by Lang (1995), the 20NG contains approximately 20,000 documents derived from 20 different newsgroups, identified by the category they regard. For each newsgroup, there are about 1,000 documents, each of which belongs to one newsgroup. Only a small fraction of documents belong to more than one newsgroup, but they were not included in the analysis. The collection topics are related to computers, politics, religion, sports and science.

Table 4.7 shows the total number of articles and unique words per category, each of which refers to one newsgroup. It should be noted that some categories are related to each other. For example, the categories "comp.graphics" and "comp.windows.x" are very similar. The same is for the categories "rec.sport.hockey" and "rec.sport.baseball" (Albishre, Albathan, & Li, 2015). Thus, the collection could be divided into 5 or 6 topics, resembling in this way both the topics it concerns and the groups of newsgroups, respectively.

In contrast to the BBC news articles collection, here only the body of each news was considered, as usually done in the literature.

Moreover, different versions of the dataset are available to use. In this work, the training subset version was used, which is composed of 11,314 documents (9,829 after the duplicates and spurious texts removal).

Table 4.7. Number of documents and unique words for each category of the 20 Newsgroups dataset (after the duplicates and spurious texts removal). The categories are divided based on the group of newsgroups they belong to.

Category	Documents	Unique words
rec.autos	492	7,467
rec.motorcycles	477	7,617
rec.sport.hockey	525	8,960
rec.sport.baseball	478	6,393
comp.sys.mac.hardware	501	6,001
comp.graphics	504	9,283
comp.windows.x	526	9,803
comp.sys.ibm.pc.hardware	538	6,570
comp.os.ms-windows.misc	486	7,789
sci.space	522	11,886
sci.med	533	12,582
sci.electronics	517	7,623
sci.crypt	541	11,339
talk.politics.guns	493	11,595
talk.politics.mideast	501	13,636
talk.politics.misc	409	10,048
talk.religion.misc	311	8,766
alt.atheism	408	8,627
soc.religion.christian	559	11,339
misc.forsale	508	7,622

4.4.2 Reuters-21578 dataset

The Reuters-21578 collection[3] contains 13,484 documents (13,190 after duplicates and spurious texts removal). Still, due to the imbalance of each category, only the largest 8 categories (the ones composed of more than 200 documents) were retained, leaving for the analysis only 8,970 documents in total (Guangxu, Yaliang, Wayne Xin, Jing, & Aidong, 2017). The collection topics are labelled as "grain", "earn", "acq", "trade", "ship", "crude", "money-fx" and "interest".

The headlines and the body of the news were included in the analysis. Moreover, as for the 20NG dataset, some topics are related, such as "trade" and "money-fx". Actually, all the Reuters categories are associated with a financial topic, making it challenging to identify a priori a reasonable number of topics to discover.

Nevertheless, it could be reasonable to suppose that the algorithm will find a number of communities equal to 6 (by clustering in a group "trade" and "money-fx" and in another group "acq" and "earn") or 5 (by clustering in a group "trade", "money-fx" and "interest" and in another group "acq" and "earn").

Table 4.8 shows the number of articles and unique words per topic.

4.4.3 20NG and Reuters results evaluation

The results obtained for the 20NG and the Reuters datasets are partially consistent with those obtained on the BBC collection, with some of the design choices made for that dataset also being robust in

[3] www.daviddlewis.com/resources/testcollections/reuters21578/

Table 4.8. Number of documents and unique words for each topic of the Reuters-21578 dataset (after the duplicates and spurious texts removal).

Topic	Documents	Unique words
grain	530	5,476
earn	3,904	8,402
acq	2,410	12,514
trade	447	6,451
ship	203	4,234
crude	524	6,996
money-fx	633	6,047
interest	319	4,251

this case.

From the inspection of Figures 4.7, 4.8 and 4.9 and Figures 4.10, 4.11 and 4.12, respectively, it is possible to observe that the number of communities found by the algorithms decreases when the window size increases, becoming stable for window sizes equal to or higher than 5. As regards the community detection algorithms employed, the Louvain algorithm performs better than the other two algorithms (especially for the SLPA algorithm). Moreover, there seem to be no differences related to the text cleaning approach or the weighting scheme applied. Finally, also for these two datasets, the results obtained by running the algorithms several times (15 times) are very similar to each other.

Therefore, based on that consideration, the results obtained in the first run were employed for the following analyses.

Interestingly, it seems that in the Reuters dataset, the possible correct number of communities is found applying the experimental

conditions 6.* for window sizes higher than 5 (Figure 4.11). In comparison, in the groups of experiments from 1.* to about 5.* the number of communities found more frequently is equal to 4. Conversely, in the 20NG dataset the possible correct number of communities is found more frequently in the groups of experiments from 1.* to 4.* for window sizes equal to 15 and 20 (Figure 4.8).

It is worth recalling that in both datasets some topics are correlated, a feature that could affect the results in terms of discovered topics.

Regarding the topic coherence measures, the scores obtained using degree, betweenness, eigenvector and closeness centrality measures to rank the words inside the communities found by using the Louvain algorithm with a window size equal to 20 on all the experimental conditions are shown in Figures 4.13 and 4.14 for the 20NG dataset and the Reuters dataset, respectively.

As for the BBC news article collection, it is possible to recognise some patterns in the coherence scores computed on the two datasets.

About the 20NG dataset, what emerges is that the last two groups of experiments show better results. This consideration lies in the evaluation of both the coherence scores and the number of communities found. Instead, the experimental conditions *.14 and *.16 show the lowest scores for almost every group of experiments. These experimental conditions are characterised by the PoS preprocessing and no filter based on the *tf-idf*.

On the contrary, the experimental condition with the highest results in two of the three coherence measures, namely UCI and NPMI, is experimental condition 1.12, for which the number of communities found is equal to 6, resembling the number of newsgroups groups. This experimental condition is characterised by the PoS preprocessing, the removal of words with a *tf-idf* value lower than 0.1, the proximity weighting scheme and no filter on the word co-occurrence matrix.

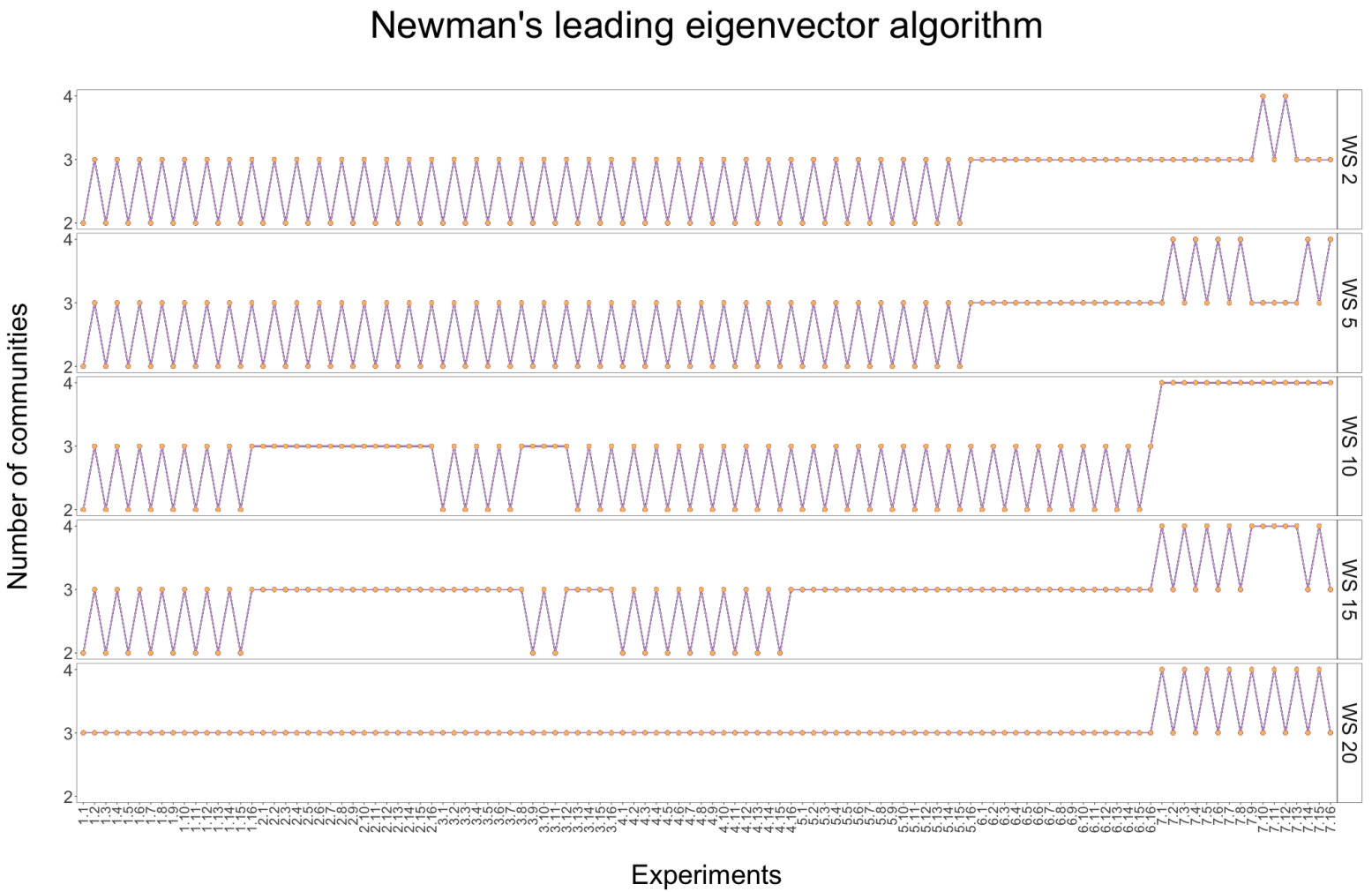


Figure 4.7. Number of communities found by the Newman's leading eigenvector algorithm per window sizes for all the experimental conditions on the 20 Newsgroups dataset. On the y-axis is the number of communities found, while on the x-axis are the experimental conditions. All the algorithms were run 15 times with each line representing a run. Here, "WS" means "window size".

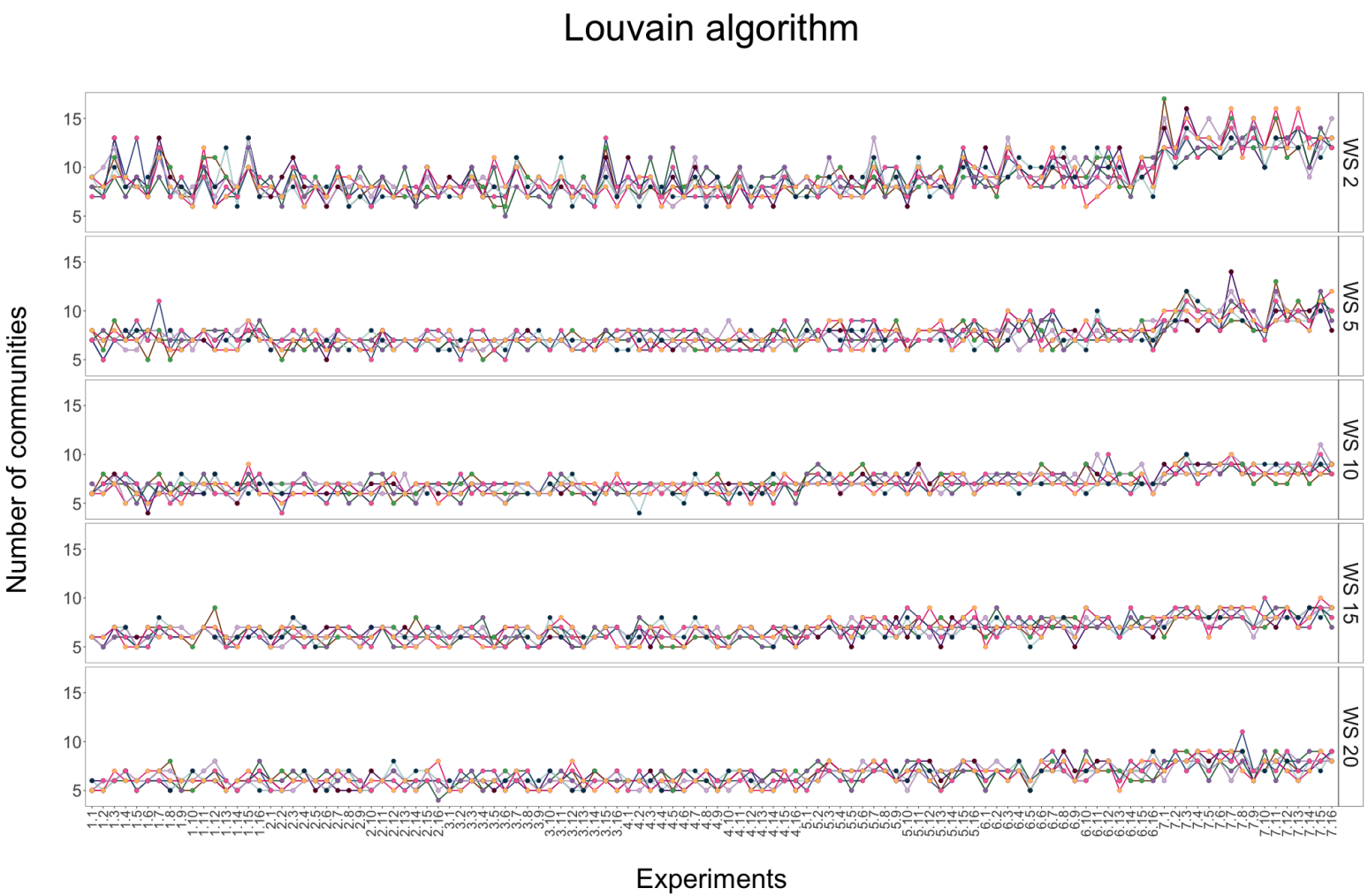


Figure 4.8. Number of communities found by the Louvain algorithm per window sizes for all the experimental conditions on the 20 Newsgroups dataset. On the y-axis is the number of communities found, while on the x-axis are the experimental conditions. All the algorithms were run 15 times with each line representing a run. Here, “WS” means “window size”.

Figure 4.9. Number of communities found by the SLPA algorithm per window sizes for all the experimental conditions on the 20 Newsgroups dataset. On the *y*-axis is the number of communities found, while on the x-axis are the experimental conditions. All the algorithms were run 15 times with each line representing a run. Here, "WS" means "window size".

Figure 4.10. Number of communities found by the Newman's leading eigenvector algorithm per window sizes for all the experimental conditions on the Reuters-21578 dataset. On the y-axis is the number of communities found, while on the x-axis are the experimental conditions. All the algorithms were run 15 times with each line representing a run. Here, "WS" means "window size".

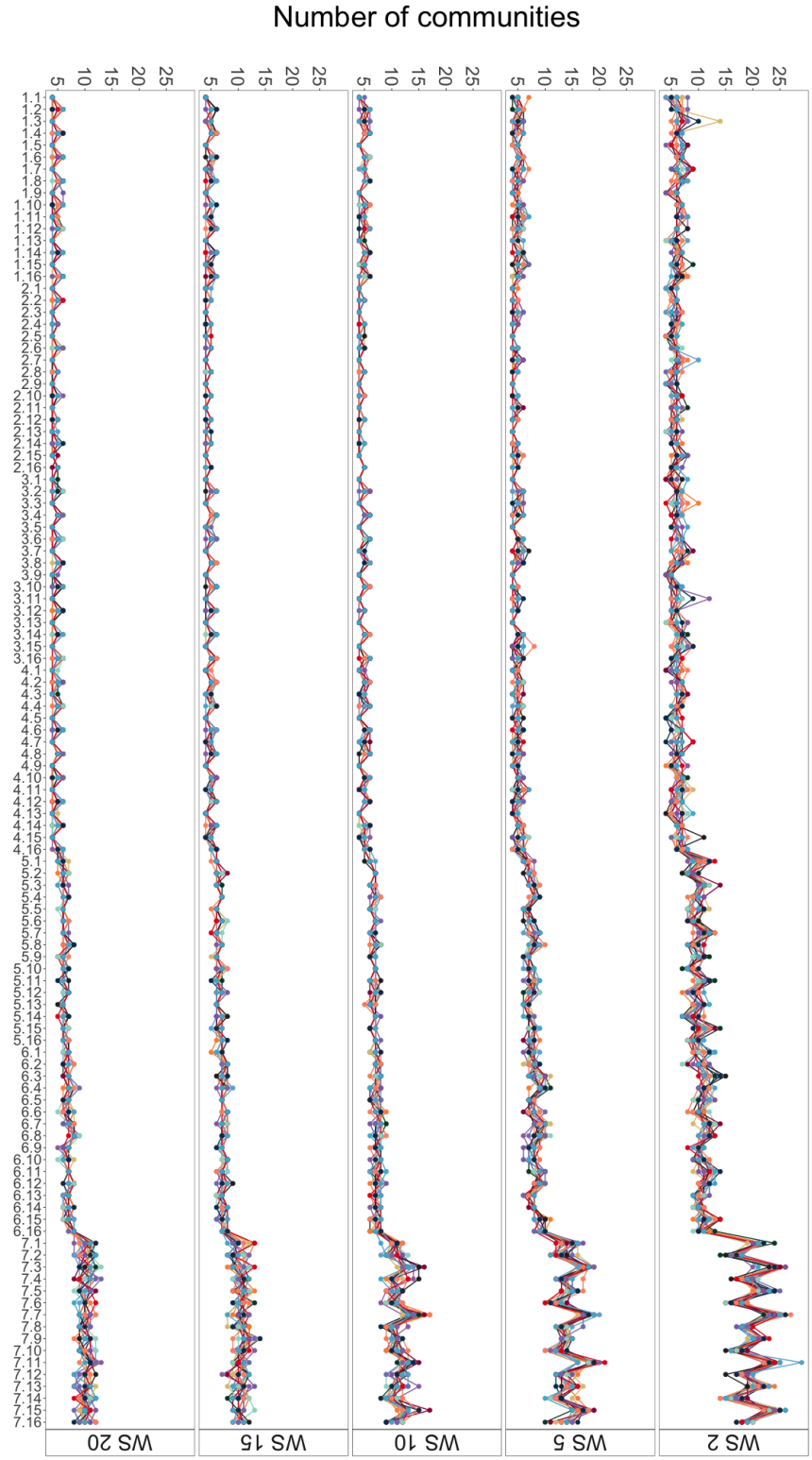


Figure 4.11. Number of communities found by the Louvain algorithm per window sizes for all the experimental conditions on the Reuters-21578 dataset. On the y-axis is the number of communities found, while on the x-axis are the experimental conditions. All the algorithms were run 15 times with each line representing a run. Here, "WS" means 'window size".

Figure 4.12. Number of communities found by the SLPA algorithm per window sizes for all the experimental conditions on the Reuters-21578 dataset. On the y-axis is the number of communities found, while on the x-axis are the experimental conditions. All the algorithms were run 15 times with each line representing a run. Here, "WS" means "window size".

However, from a manual inspection of the communities found in the experimental condition 1.12, it emerged that its communities are very unbalanced, with the last community composed of only 5 words. For this reason, it was decided to use for the subsequent analyses the results of the experimental condition 7.1, which presents the highest UMass score and for which the communities are pretty balanced. This experimental condition is characterised by the normal preprocessing, the removal of words with a *tf-idf* value lower than 0.01, the weighted weighting scheme and the removal of the top 500 words from the word co-occurrence matrix. Note that also in this case the number of communities found is equal to 6. The coherence scores of the experimental condition 7.1 are shown in Table 4.9.

As regards the Reuters dataset, results show a pattern similar to the 20NG one. In particular, the first and the last experimental conditions in almost all groups of experiments show the lowest coherence scores, while the experimental conditions from *.11 to *.14 for the first four groups of experiments exhibit the highest scores. However, the groups of experiments 7.* and 6.* have the highest coherence scores. Indeed, the experimental condition with the highest UMass score is experimental condition 6.8, while experimental condition 7.6 is the one with the highest UCI and NMPI scores. In both experimental conditions, the number of communities found is equal to 8.

However, as for the 20NG dataset, from a manual inspection of the communities found in the experimental conditions 6.8 and 7.6, it comes to light that the communities founds are not balanced, with communities 7 and 8 composed of just a few words. For this reason, in order to find a balance between the communities' composition and coherence scores, experimental condition 6.2 was chosen for the subsequent analyses (see Table 4.9 for its coherence scores). This

experimental condition is characterised by the PoS preprocessing, the removal of words with a *tf-idf* value lower than 0.01, the weighted weighting scheme and the removal of the top 100 words from the word co-occurrence matrix.

4.5 Comparison with probabilistic topic models

To conclude, topic coherence metrics and precision, recall and F1-score measures were computed on all the datasets using the network-based approach and classical topic models.

About the topic models, the correct number of topics was given to them a priori in order to obtain the best possible results. Table 4.9 shows the topic coherence scores obtained on the BBC news article collection, the 20NG dataset and the Reuters dataset using the network-based approach and the classical topic models. The table shows that, in general, the network-based approach outperforms LDA, NMF and LSI in UCI and NPMI measures while obtaining lower results than BERTopic and BERTopic-MPNET for the UMass coherence metric. However, it should be noted that for BERTopic and BERTopic-MPNET it has been necessary to use K-means instead of HDBSCAN (used by default) as clustering methods due to the high level of documents classified as irrelevant or outliers.

Regarding the clustering performance indicators, the results are reported in Table 4.10. Also in this case, from the inspection of the table, it is possible to observe that the network-based approach outperforms LDA, NMF and LSI in terms of precision, recall and F1-score while showing lower results than the ones obtained using BERTopic and

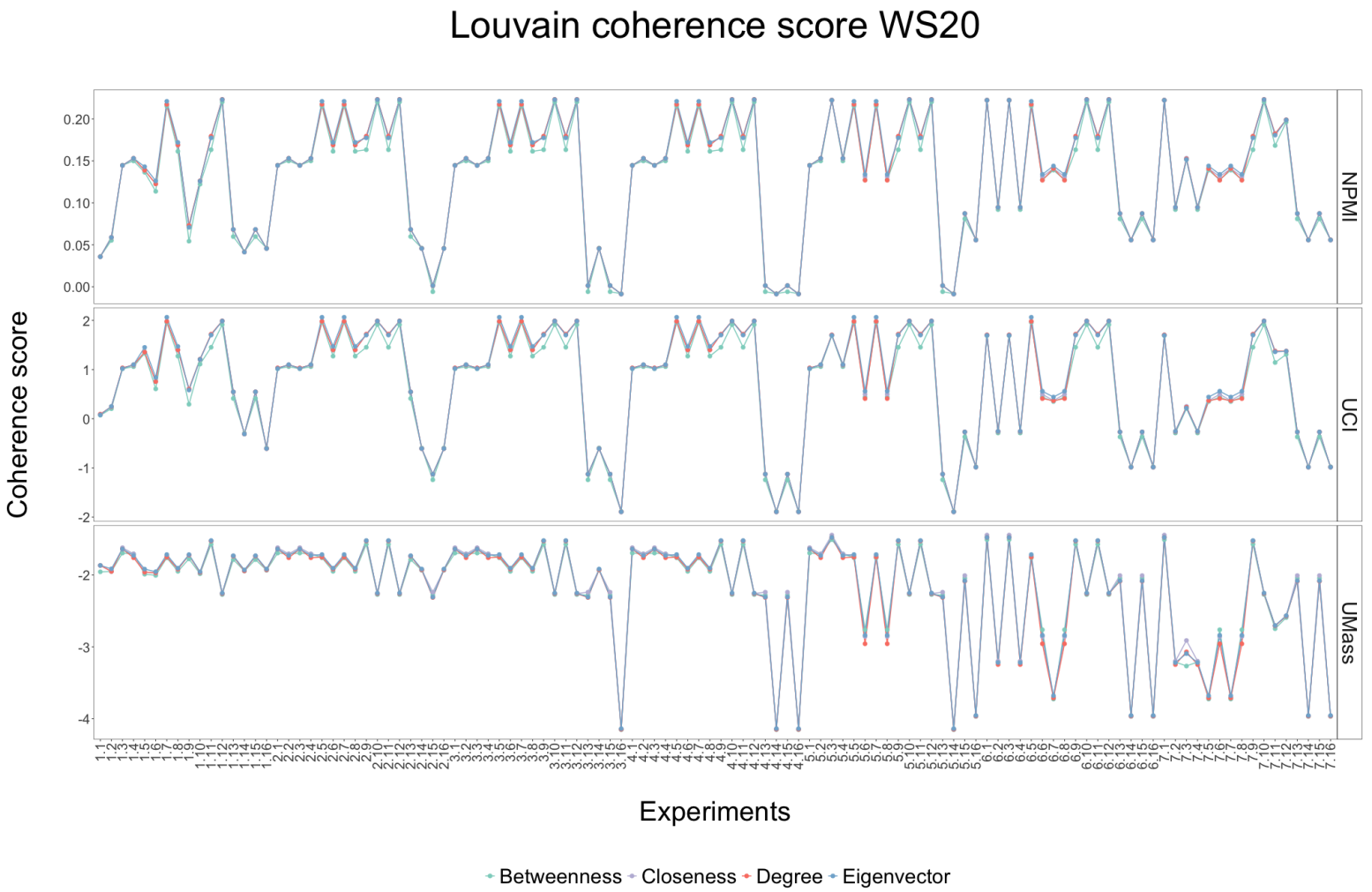


Figure 4.13. Topic coherence scores for all the experimental conditions using the Louvain community detection algorithm for a window size equal to 20 on the 20 Newsgroups dataset. On the y-axis are reported the topic coherence scores, while the x-axis are the experimental conditions. Note that the y-axis has different scales for each coherence method.

Figure 4.14. Topic coherence scores for all the experimental conditions using the Louvain community detection algorithm for a window size equal to 20 on the Reuters-21578 dataset. On the y-axis are reported the topic coherence scores, while the x-axis are the experimental conditions. Note that the y-axis has different scales for each coherence method.

BERTopic-MPNET on the BBC news article collection. However, in judging this result, it should recall that for BERTopic and BERTopic-MPNET it is necessary to define a priori the number of topics.

About the 20NG dataset, the results of the classical topics models are consistent with those reported in the literature (for example, see Guangxu et al., 2017), while the network-based approach outperforms the other methods. Regarding the Reuters dataset, the performances of all methods are very similar to each other, with the LDA and the network-based approach working better than the others.

From these results, it can be stated that the network-based approach, while giving lower results than the embedding models on datasets with clear-cut topics, obtains better or similar results to classical topic models on datasets with correlated topics.

Table 4.9. Topic coherence scores for the BBC news article collection, the 20NG dataset and the Reuters dataset using both the network-based approach and classical topic models. In bold the highest results per row.

	Coherence measures	LDA	LSI	NMF	BERTopic	BERTopic MPNET	Network approach
BBC	UMass	-1.61	-1.70	-1.98	**1.00**	**1.00**	-1.41
	UCI	-0.25	0.28	-0.33	-0.14	-0.14	**0.41**
	NPMI	0.02	**0.06**	0.03	-0.03	-0.03	**0.06**
20NG	UMass	-1.88	-2.07	-2.33	**1.00**	**1.00**	-1.48
	UCI	-0.06	0.69	-0.17	-0.12	-0.11	**1.70**
	NPMI	0.03	0.11	0.03	-0.03	-0.02	**0.22**
Reuters	UMass	-1.63	-1.66	-1.94	**-0.05**	-0.07	-1.22
	UCI	0.25	0.42	-0.11	0.66	0.69	**2.58**
	NPMI	0.06	0.08	0.04	0.17	0.12	**0.27**

Table 4.10. Average performance indicator scores for the BBC news article collection, the 20NG dataset and the Reuters dataset using both the network-based approach and classical topic models. In bold the highest results per row.

	Coherence measures	LDA	LSI	NMF	BERTopic	BERTopic MPNET	Network approach
BBC	Precision	0.76	0.70	0.85	0.96	**0.97**	0.92
	Recall	0.77	0.29	0.80	0.96	**0.97**	0.91
	F_1-score	0.72	0.21	0.81	0.96	**0.97**	0.92
20NG	Precision	0.48	0.21	0.38	0.56	0.53	**0.80**
	Recall	0.43	0.24	0.16	0.50	0.42	**0.59**
	F_1-score	0.40	0.19	0.08	0.45	0.45	**0.60**
Reuters	Precision	0.55	0.41	0.33	0.43	0.48	**0.57**
	Recall	**0.54**	0.50	0.24	0.50	0.52	0.42
	F_1-score	**0.47**	0.34	0.16	0.40	0.46	0.42

Chapter 5

Empirical application on LexisNexis news database

To assess the quality of the network-based approach, this chapter presents its application to a news articles collection about distance learning published from 1st March 2020 to 31st May 2020 in four of the most important Italian newspapers: the Corriere della Sera, il Resto del Carlino, Il Giorno and La Nazione.

The data are included in the LexisNexis news database, an online platform that collects European and worldwide legal opinions, public records, news and business information.

In a previous work (C. Galluccio, Crescenzi, & Petrucci, 2021), the Italian newspapers' narratives about distance learning during the first wave of the COVID-19 pandemic were investigated using the same data and employing, among others, the LDA model.

Here, the aim is to compare the results obtained in that work with those found using the network-based approach.

5.1 The Italian newspapers' narrative on distance learning during the COVID-19 pandemic

In March 2020, the World Health Organization (WHO) officially declared the COVID-19 outbreak a pandemic[1]. Consequently, governments worldwide had to introduce several protective measures to contain the spread of the virus, including the temporary closure of educational institutions of all grades. Teachers had to act accordingly, urgently adopting distance learning for all their classes in order to maintain educational continuity (Dietrich et al., 2020).

In Italy, school and university closures started on March 4th, thus going ahead until the end of the school year. Before the sanitary emergency, teachers felt some resistance to distance learning. Indeed, as stated in Galdieri (2020), teaching is seen as something that takes place almost exclusively in person: space and time are the fundamental elements that characterise teaching, which occurs primarily in classes and schools that, together with the temporal structure, have strongly influenced teaching organisation. Consequently, to be "genuinely" educational, every environment must be organised spatially and temporally, analysing the dynamics that regulate the teaching-learning process and educational-didactic communication.

Conversely, distance learning can be seen as a tool whose boundaries (temporal and geographical) remain uncertain: on the one hand, it creates an evident physical distance (although it has often required

[1] https://www.who.int/director-general/speeches/detail/who-director-general-s-opening-remarks-at-the-media-briefing-on-covid-19---11-march-2020

students to share home spaces with the rest of the family); on the other hand, it determines a change in terms of school time (for instances, not entirely sustainable at a distance, especially for primary school; Palareti, 2020; Roncaglia et al., 2021).

In this vein, the COVID-19 pandemic has exhibited its purely social (rather than natural) connotations, highlighting our unpreparedness as a society for the digital tools with which, every day, perhaps unwittingly, we share our identity. This unpreparedness arises not only in terms of ownership of the technological tools but also in the inability to use them adequately. In this regard, numerous studies have investigated the satisfaction of teachers, students, and families in distance learning (see Bacci, Fabbricatore, & Iannario, 2022; Galdieri, 2020; Lucisano, 2020; Roncaglia et al., 2021).

In this context, studying newspapers' narratives about distance learning can help understand how newspapers reported it and their potential influence on readers' perceptions. Indeed media play a fundamental role in shaping public opinion about events with respect to which people's attention and sensitivity have increased, also as a consequence of the COVID-19 sanitary emergency. Several studies have highlighted the central role of media in forming public opinion and influencing people's perception of events in different fields (see, for example, Garz, 2012; Tausch & Zumbuehl, 2018). In this regard, the agenda-setting theory is one of the most important theories used to explain how media attempts to influence people's perception, choosing and displaying particular topics and consequently affecting the salience attitude towards specific issues (McCombs & Shaw, 1972).

About distance learning, as reported in Kuzelj and Šamija (2020), the newspapers' narrative about educational issues can contribute to the development of a better-informed public debate. This a significant issue since school digitalisation does not only consist of purchasing

technological instruments or equipment; instead, it consists of not considering digital learning as an alternative to face-to-face learning but one of its dimensions (Rivoltella, 2020). In other words, it means to understand how to change without degenerate (Galdieri, 2020).

For this reason, the Italian newspapers' narratives about distance learning during the first wave of the COVID-19 pandemic (from March to May 2020) were investigated. To this end, the network-based approach was applied to the news about distance learning published in four of the most important Italian journals available in the LexisNexis news database, namely the Corriere della Sera, il Resto del Carlino, Il Giorno and La Nazione, comparing the results with the one obtained using the LDA model (Blei et al., 2003).

5.2 Data description and text preprocessing

Newspapers are one of the primary sources through which Italians get information: in March 2020, about 2,5 million copies of newspapers were sold[2], and it was found that in the first quarter of 2020, around 30% of Italians have read every day both printed and digital newspapers[3]. In this regard, ISTAT reported that in 2020 about 31.2% of Italians had

[2] https://www.agcom.it/documents/10179/19412594/Documento+generico+15-07-2020/46c0ae2c-dff7-4e26-826c-09938ce376fb?version=1.0

[3] http://audipress.it/pubblicati-dati-audipress-2020-91-la-lettura-digitale/

read a newspaper at least once a week[4]. Moreover, it is worth noting that every newspaper's content is also shared online, primarily through social media (such as Facebook, Twitter and Instagram), increasing the visibility of published news. For this reason, it was considered reasonable to analyse newspapers' narratives on distance learning and their potential impact on public opinion.

Hence, to study the Italian newspapers' narrative about distance learning during the first wave of the COVID-19 pandemic, a study on the four Italian newspapers mentioned above included in the LexisNexis news database, an online platform that collects European and worldwide legal opinions, public records, news and business information[5] was carried out. More specifically, the news regarding distance learning published from 1st March 2020 to 31st May 2020 in the Corriere della Sera, il Resto del Carlino, Il Giorno, and La Nazione was included in the analysis.

Regarding the newspapers, the Corriere della Sera is the leading Italian daily by circulation, ranking second in readership and copies sold. About the other three newspapers, il Resto del Carlino, based in Bologna and distributed mainly in Emilia-Romagna, Marche, and Rovigo, ranks fifth for the number of readers, while La Nazione, printed and edited in Florence, and Il Giorno, published in Milan, rank respectively tenth and twenty-fifth for the number of readers[2].

The first step of the data preprocessing phase was the extraction of the news of interest. To filter the news, selected keywords on news headlines and bodies were employed. Keywords were chosen considering the most common words generally used to refer to distance learning in

[4] http://dati.istat.it/

[5] https://www.lexisnexis.com/communities/academic/w/wiki/30.lexisnexis-academic-general-information.aspx

the literature. For example, news whose headline or body included the terms “dad” (Italian acronym for distance learning) or “didattica a distanza” (distance learning) was included in the analysis[6].

The resulting corpus was then preprocessed by removing non-alphanumeric characters, numbers and stopwords. Afterwards, the text was tokenised, choosing the single word as the unit of analysis. Then, a stemming algorithm was applied.

5.3 Text analysis of Italian newspapers

The total number of articles published from 1st March 2020 to 31st May 2020 about distance learning in the Corriere della Sera, il Resto del Carlino, Il Giorno and La Nazione is reported in Table 5.1. It is worth noting that each newspaper has published approximately the same number of news on distance learning (between 3.2% and 3.4% of the total).

Then, the network-based approach was applied to understand the main arguments discussed in the collection. In particular, the analysis was carried out following the three paths that provide the best results in the datasets investigated in Chapter 4. Therefore, the experimental conditions 1.1, 6.2 and 7.1 were applied to the articles about distance

[6]The complete list of keywords used to filter news is: “dad”, “didattica a distanza”, “scuola 2.0”, “didattica”, “scuola digitale”, “lezioni online", “didattica online", “lezioni digitali", “digitalizzazione della scuola", “lezioni web", “scuola vera", “scuola a distanza", “scuola del futuro", “istruzione", “scuola online", “lezioni a distanza", “insegnamenti telematici", “scuola e digitale", “diseducazione digitale", “educativ", “socio-educativ", “scuola e futuro", “lezioni on-line", “scuola on-line", “lezioni on line", “scuola on line", “lezioni web", “scuola web", “lezioni a distanza", “classe web"

Table 5.1. Number of articles published from March to May 2020 about distance learning in the Corriere della Sera, il Resto del Carlino, Il Giorno and La Nazione.

Newspapers	Articles published from March to May 2020	Articles about distance learning
Corriere della Sera	20167	694
il Resto del Carlino	64555	2222
Il Giorno	27141	882
La Nazione	63859	2083

learning included in the LexisNexis dataset.

While sharing the weighting scheme and the filter based on the *tf-idf*, these experimental conditions differ in the text cleaning approach and the filter on the word co-occurrence matrix. Moreover, consider the results obtained in Chapter 4, in this case only the window size equal to 20 and the Louvain algorithm were used.

5.4 Evaluation of detected topics

The number of communities found in experimental conditions 1.1, 6.2 and 7.1 was equal to 5, 9 and 13, respectively. However, from a manual inspection of the communities found, it emerged that in the last two experimental conditions, there were 5 bigger communities, with the others being just residual. For this reason, only the results obtained in experimental condition 1.1 were investigated.

Table 5.2 shows the top 15 words with the highest node degree centrality measure under each community found in the experimental condition 1.1, while Table 5.3 shows the top 15 word probabilities

under the 3 topic inferred via LDA (the number of topics chosen a priori was determined after different tries and the evaluation of the results; C. Galluccio et al., 2021).

Looking at Table 5.2, it is possible to recognise most of the central issues discussed in the news about distance learning in the period considered. More specifically, from the inspection of the table emerges that Community 2 concerns two fundamental aspects of distance learning: its social aspects (related to the sanitary emergency) and the role of families.

Regarding the former, the COVID-19 sanitary emergency had enormous consequences on the social, economic and cultural system. The educational and cultural spheres were among those most directly affected by the pandemic due to the closure of schools, universities, and cultural services and activities. Hence, the effects of the emergency were dramatic, albeit in different forms and ways, on the majority of cultural sectors (Roncaglia et al., 2021).

For these reasons, it seems reasonable to suppose that journals' narratives about distance learning also regard its social aspects related to the emergency. Somehow, the same distance learning could be defined more as "emergency learning" rather than "distance learning" (Roncaglia, 2020).

Additionally, distance learning had repercussions on families too. Indeed, as in many countries worldwide, Italian schools and families from all socio-cultural backgrounds had to reorganise teaching-learning paths to guarantee educational continuity. In this vein, distance learning has crossed the family line, tearing the boundaries between the school-family educational agencies, disrupting rules and actions that guarantee them an identity and a clear role in respect and mutual commitment (Roncaglia et al., 2021).

Similar insights can be deduced by analysing Community 4, which

Table 5.2. Top 15 words with the highest node degree centrality measure under each community found in the experimental condition 1.1 in the LexisNexis dataset. In parenthesis, the degree centrality score is rounded to two decimals.

Community 1	Community 2	Community 3	Community 4	Community 5
cas (0.45)	comun (0.57)	scuol (0.62)	don (0.32)	ministr (0.43)
port (0.33)	serviz (0.54)	didatt (0.56)	san (0.31)	president (0.41)
ital (0.32)	attiv (0.54)	lezion (0.55)	consegn (0.26)	istruzion (0.39)
raccont (0.30)	lavor (0.53)	student (0.54)	marc (0.22)	situazion (0.37)
viv (0.30)	educ (0.49)	distant (0.52)	acquist (0.22)	cont (0.36)
arrive (0.30)	famigl (0.48)	docent (0.45)	mascherin (0.21)	azzolin (0.35)
coronavirus (0.30)	emerg (0.45)	ragazz (0.42)	sant (0.21)	chied (0.35)
pens (0.29)	pubblic (0.43)	insegn (0.41)	ferm (0.20)	esam (0.35)
piccol (0.29)	progett (0.41)	istit (0.40)	raccolt (0.19)	dat (0.35)
pass (0.29)	centr (0.41)	spieg (0.38)	paol (0.19)	settembr (0.34)
tant (0.28)	social (0.41)	iniz (0.36)	volontar (0.18)	prov (0.34)
trov (0.27)	scolast (0.40)	onlin (0.36)	protezion (0.18)	risc (0.33)
parl (0.27)	bambin (0.40)	class (0.36)	alessandr (0.17)	problem (0.33)
figl (0.26)	eur (0.40)	possibil (0.34)	donazion (0.17)	punt (0.33)
ser (0.26)	comunal (0.39)	cors (0.32)	distribu (0.17)	luc (0.31)

is related to the problems subsequent to the start of the pandemic, with words related to voluntary (religious) associations and sanitary tools needed to face the virus (e.g. the word "mascherin" which means "face mask").

Not surprisingly, Community 3 regards distance learning and all its features. Thus, it is possible to observe words such as "scuol" (school), "didatt" (didact), "student" (student), and "lezion" (lesson), but also words such as "distant" (distance) and "onlin" (online).

Community 5 clearly regards a political aspect of distance learning, with words related to the government in general. For example, one of the words with the highest node degree centrality measure is "azzolin", namely the last name of the school minister in March 2020 "Azzolina". What is interesting in this topic is the hint regarding a "temporal" component of distance learning, shared somehow by Community 1. This finding is supported by the presence of words such as "settembr" (September) in Community 5, and "raccont" (narrate), "arrive" (reach) and "piccol" plus "pass" ("piccoli passi", which means reaching something, like a big task, step by step) in Community 1. In this regard, the sanitary emergency and the introduction of distance learning have changed the temporal organisation of both schools and our lives in at least three different ways. First, during the period under analysis, people wondered when the sanitary emergency would end, an issue deeply felt during the first wave of the COVID-19 pandemic. Secondly, concerning the school, as aforementioned, the sanitary emergency has upset the time of the school in a practical way (for example, by modifying school time). Furthermore, it is worth noting that these changes have affected students and teachers, for example, with the latter being compelled to work harder to create online content for their classes. Finally, the COVID-19 pandemic has forced to accelerate

school digitalisation and has represented a breaking point between the “school of yesterday" and the “school of tomorrow". In this vein, the words that emerged in these communities revoke these concerns, both regarding the uncertainty about the subsequent academic year and the end of the social restrictions. Indeed, these aspects were deeply felt by the population, which on the one hand, was exhausted of the social restrictions and the distance learning workload, whereas, on the other hand, aimed to “return to normality”.

Comparing these results with the ones obtained using the LDA model (Table 5.3), it emerges that 3 of the 5 communities found using the network-based approach resemble the topics inferred via LDA. More specifically, these topics are:

- Community 1, which refers to a “temporal” component of distance learning;
- Community 2, which concerns the role of families and the social aspects related to distance learning;
- Community 3, which regards distance learning and all of its features.

Therefore, in addition to that, through the network-based approach, it was possible to discover two additional communities: Community 4, which is related to the problems subsequent to the start of the pandemic, with words related to voluntary (religious) associations and sanitary tools needed to face the virus; Community 5, that clearly regards a political aspect of distance learning, with words related to the government in general, as said before.

From these results, it is possible to state that the COVID-19

Table 5.3. Top 15 word probabilities under a 3 topic inferred via LDA.

Topic 1		Topic 2		Topic 3	
word	beta	word	beta	word	beta
comun	0.0123	scuol	0.0281	piu	0.0115
famigl	0.0119	didatt	0.0137	cas	0.0098
attiv	0.0117	distanz	0.0135	giorn	0.0074
serviz	0.0111	student	0.0124	cos	0.0069
educ	0.0104	lezion	0.0112	stat	0.0069
lavor	0.0074	ragazz	0.0083	anni	0.0059
bambin	0.0071	piu	0.0081	sol	0.0054
progett	0.0066	scolast	0.0077	via	0.0050
mil	0.0061	insegn	0.0072	prim	0.0049
emergent	0.0060	far	0.0066	perc	0.0047
centr	0.0060	class	0.0065	far	0.0046
pubblic	0.0059	docent	0.0065	temp	0.0044
eur	0.0058	anno	0.0059	tant	0.0042
stat	0.0058	istit	0.0059	molt	0.0041
social	0.0055	onlin	0.0054	grand	0.0041

sanitary emergency has undoubtedly changed the international political and economic scenarios and affected social and relational dynamics, thus showing its nature as both a health and social emergency.

The health emergency has caught many teachers unprepared to face distance learning in the school context because of their lack of experience in ICTs technologies (usually limited to IWB, tablet, or electronic register). Additionally, practical problems arose, such as the possibility of accessing technological tools and infrastructural facilities. Consequently, if improvised and unstructured, distance learning can lead to several issues directly affecting teachers, students, and their families.

For these reasons, in this context, media played a fundamental role as sources of information. Therefore, these findings take on a significant

value regarding distance learning because understanding how the media represented this tool and how people could consider it can help develop an efficient strategy to facilitate the school digitalisation process. An informed discussion on these topics allows people to address the several aspects of this digital revolution critically arising, albeit in a forcibly accelerated way, by the pandemic.

Conclusions

In the context of textual analysis, network-based procedures for topic detection are gaining attention, also as an alternative to classical topic models.

Network-based procedures are based on the idea that documents can be represented as word co-occurrence networks, where topics are defined as groups of strongly connected words. Although many works have used network-based procedures for topic detection, there is a lack of systematic analysis of how different design choices, such as the building of the word co-occurrence matrix and the selection of the community detection algorithm, affect the final results in terms of detected topics. Another unexplored question about network-based topic detection concerns its relationship with classical topic models, such as the LDA model.

Therefore, the aim of this thesis was to address these questions by developing a deeper understanding of optimal design choices for network-based procedures for topic detection, showing how and to what extent the choices made during the design phase affect the results, and contextually comparing these procedures with classical topic models. Then, the network-based procedure was applied to a real-world dataset to show its effectiveness when studying social phenomena through which the attention and sensitivity of public opinion are oriented.

To answer the research questions postulated in Section 1:

RQ1: ***Could the text preprocessing and, consequently, the keyword selection affect the results regarding the features of the discovered topics?***

About the effects of the text preprocessing and the keywords selection on the features of the detected topics, results showed that the text cleaning approach seems to affect the results depending on the characteristics of the datasets, delivering better results when applied to datasets with highly correlated topics.

Furthermore, the results regarding the number of communities found concerning the text preprocessing used are interesting. In fact, the number of communities found by the three algorithms remains stable in the different experimental conditions, even if the vocabulary size is smaller when cleaning the text using a PoS tagger. However, this choice seems to affect the results regarding topic coherence, with the PoS preprocessing performing, even if only slightly, better than the classic preprocess.

Finally, removing words with a *tf-idf* value lower than 0.01 showed to be a robust design choice when applying the network-based approach for topic detection.

RQ2: ***Does manipulation in defining the word co-occurrence matrix has an impact on the quality of the discovered topics?***

The findings show that, for all tested algorithms, increasing the window size initially decreases the number of communities, which becomes stable for window sizes equal to or greater than

5, depending on the algorithm. This suggests that some of the topics identified in the literature may have been influenced by this design choice, leading to the consideration that the window size should be regarded as an important hyperparameter in this kind of study.

Regarding the filter on the word co-occurrence matrix, the results show that while removing the less common words can improve the results, giving the chance to find optimal results with a smaller vocabulary size, finding a good setting for applying the removal of a common word is not straightforward. In fact, while the removal of common words (not removed by the filter on the *tf-idf* value) negatively affects the results in the case of a good-quality dataset with clear-cut topics, it seems to lead to better results when applied to poor-quality datasets with correlated topics.

Finally, for all the datasets, the weighted weighting scheme showed to be, as well as the removal of words with a *tf-idf* lower than 0.01, a robust design choice in this kind of analysis.

RQ3: ***Is the detected topics' number and content influenced by the community detection algorithm chosen? If so, to what extent?***

Considering the number of detected topics applying different filters on the word co-occurrence matrix, it is possible to observe that the Louvain community detection algorithm generally performs better than the other tested algorithms. For example, considering the information available on the number of topics in the BBC documents collection, the Louvain algorithm always detects the correct number of topics for window sizes greater

than 5, whereas the other two algorithms fail.

This does not lead to a rejection of the hypothesis that overlapping community detection methods are more appropriate to find topics in word co-occurrence networks: it is still possible that the Louvain algorithm could correctly cluster together words belonging to a single topic while arbitrarily including multi-topic words in only one of the communities where they should have been included.

However, it is eventually possible to conclude that some of the typical overlapping community detection methods are not able to identify significant topics under the experimental settings tested in this work. Note that the fact that these settings are taken from the literature suggests that more research should be done to identify preprocessing schemes leading to networks better suited to applying these methods. One feature of the networks obtained in these experiments that may have determined the poor results of the tested methods is their high density, suggesting that stronger filtering schemes should be considered.

RQ4: ***When analysing text news data, could the network-based procedure for detecting topics represent a valid alternative to classical topic models?***

Both for the topic coherence measures and the clustering performance metrics, the network-based approach outperforms LDA, NMF and LSI while showing lower results than the ones obtained using BERTopic and BERTopic-MPNET on almost all the datasets.

Nevertheless, in judging this result, it should recall that for

BERTopic and BERTopic-MPNET, as for the other topic model methods, it is necessary to define a priori the number of topics and that, in this case, the correct number of topics was selected directly. However, knowing the right number of topics in advance is not so straightforward, but usually requires several tries and results inspections.

Conversely, in the network-based procedures for topic detection, there is no need to choose a priori the number of topics, representing in this way a noticeable advantage for the researcher, also considering their good results in terms of both topic coherence and clustering.

Finally, starting from these results, it can be stated that the network-based approach, while giving lower results than the embedding models on datasets with clear-cut topics, obtains better or similar results than classical topic models on correlated topics.

Then, the application of the network-based procedure to the Italian news articles about distance learning also showed interesting results, recognising the central issues discussed in the news about distance learning in the period considered. In particular, the network-based approach delivers two additional topics initially not identified by the LDA model, which lead to a novel and interesting point of view about the news media representation of the problems related to the pandemic.

In summary, on the one hand, the results confirm what is stated in the literature, where network-based procedures for topic discovery show promising results; on the other hand, they highlight how different design choices, such as choosing specific algorithms or window sizes, applying filters on the word co-occurrence matrix, or defining different

filter based on the *tf-idf*, may significantly affect the results in terms of detected topics.

Bibliography

Aggarwal, C. C. (2018). *Machine learning for text.* Cham: Springer Nature Switzerland.

Albishre, K., Albathan, M., & Li, Y. (2015). Effective 20 Newsgroups Dataset Cleaning. In C. Da trovare (Ed.), *2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)* (Vol. 3, pp. 98–101). Los Alamitos: IEEE Computer Society.

Alcoforado, A., Ferraz, T. P., Gerber, R., Bustos, E., Oliveira, A. S., Veloso, B. M., ... Costa, A. H. R. (2022). ZeroBERTo: Leveraging Zero-Shot Text Classification by Topic Modeling. In V. Pinheiro et al. (Eds.), *Computational Processing of the Portuguese Language* (Vol. 13208, pp. 125–136). New York City: Springer International Publishing.

Alghamdi, R., & Alfalqi, K. (2015). A survey of topic modeling in text mining. *International Journal of Advanced Computer Science and Applications (IJACSA)*, *6*(1), 147–153.

Allahyari, M., Pouriyeh, S., Assefi, M., Safaei, S., Trippe, E. D., Gutierrez, J. B., & Kochut, K. (2017). A Brief Survey of Text Mining: Classification, Clustering and Extraction Techniques. *arXiv:1707.02919*, 1–17. Retrieved from https://doi.org/10.48550/arXiv.1707.02919

Amancio, D. R., Aluisio, S. M., Oliveira, O. N., & Costa, L. d. F. (2012). Complex networks analysis of language complexity. *Europhysics Letters (EPL)*, *100*(5), 1–6.

Bacci, S., Fabbricatore, R., & Iannario, M. (2022). Multilevel IRT models for the analysis of satisfaction for distance learning during the covid-19 pandemic. *Socio-Economic Planning Sciences*, 1–11.

Bernard, H. R., & Ryan, G. (1998). Text analysis: Qualitative and Quantitative Methods. In H. R. Bernard (Ed.), *Handbook of methods in cultural anthropology* (Vol. 613, pp. 595–645). Walnut Creek: AltaMira Press.

Berry, M. W., Dumais, S. T., & O'Brien, G. W. (1995). Using linear algebra for intelligent information retrieval. *SIAM review*, *37*(4), 573–595.

Blei, D. M. (2012). Probabilistic topic models. *Communications of the ACM*, *55*(4), 77–84.

Blei, D. M., Kucukelbir, A., & McAuliffe, J. D. (2017). Variational inference: A review for statisticians. *Journal of the American statistical Association*, *112*(518), 859–877.

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

Blondel, V. D., Guillaume, J., Lambiotte, R., & Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, *2008*(10), 1–12.

Bolasco, S. (2005). Statistica testuale e text mining: alcuni paradigmi applicativi. *Quaderni di statistica*, *7*, 17–53.

Borgatti, S. P., Everett, M. G., & Johnson, J. C. (2018). *Analyzing social networks*. London: Sage.

Boutsidis, C., & Gallopoulos, E. (2008). SVD based initialization: A

head start for nonnegative matrix factorization. *Pattern recognition*, *41*(4), 1350–1362.

Boyd, D., & Crawford, K. (2012). Critical questions for big data: Provocations for a cultural, technological, and scholarly phenomenon. *Information, communication & society*, *15*(5), 662–679.

Brunet, J. P., Tamayo, P., Golub, T. R., & Mesirov, J. P. (2004). Metagenes and molecular pattern discovery using matrix factorization. *Proceedings of the national academy of sciences*, *101*(12), 4164–4169.

Bullinaria, J. A., & Levy, J. P. (2007). Extracting semantic representations from word co-occurrence statistics: A computational study. *Behavior Research Methods*, *39*, 510–526.

Buntine, W. (2009). Estimating likelihoods for topic models. In Z. H. Zhou & T. Washio (Eds.), *Advances in Machine Learning* (Vol. 5828, pp. 51–64). Heidelberg: Springer-Verlag.

Butler, S., Wermelinger, M., Yu, Y., & Sharp, H. (2011). Improving the tokenisation of identifier names. In M. Mezini (Ed.), *European Conference on Object-Oriented Programming* (Vol. 6813, pp. 130–154). Berlin: Springer-Verlag.

Campello, R. J. G. B., Moulavi, D., & Sander, J. (2013). Density-based clustering based on hierarchical density estimates. In J. Pei, V. S. Tseng, L. Cao, H. Motoda, & G. Xu (Eds.), *Pacific-Asia Conference on Knowledge Discovery and Data Mining* (pp. 160–172). Heidelberg: Springer Berlin.

Chang, J., & Blei, D. M. (2009). Relational topic models for document networks. In *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics, editor = van Dyk, D. and Welling, M., pages = 81–88* (Vol. 5). Florida: Proceedings of Machine Learning Research.

Chen, Y., Zhang, H., Liu, R., Ye, Z., & Lin, J. (2019). Experimental explorations on short text topic mining between LDA and NMF based Schemes. *Knowledge-Based Systems*, *163*, 1–13.

Cios, K. J., Swiniarski, R. W., Pedrycz, W., & Kurgan, L. A. (2007). *Data Mining: A Knowledge Discovery Approach*. Heidelberg: Springer-Verlag.

Cohen, R., & Havlin, S. (2010). *Complex networks: structure, robustness and function*. Cambridge: Cambridge University Press.

Dang, T., & Nguyen, V. T. (2018). ComModeler: Topic Modeling Using Community Detection. In C. Tominski & T. von Landesberger (Eds.), *EuroVis Workshop on Visual Analytics (EuroVA)* (pp. 1–5). Goslar: The Eurographics Association.

de Arruda, H. F., Costa, L. F., & Amancio, D. R. (2015). Topic segmentation via community detection in complex networks. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, *26*(6), 1–10.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American society for information science*, *41*(6), 391–407.

de Groot, M., Aliannejadi, M., & Haas, M. R. (2022). Experiments on Generalizability of BERTopic on Multi-Domain Short Text. *arXiv:2212.08459*, 1–3. Retrieved from https://doi.org/10.48550/arXiv.2212.08459

De Nooy, W., Mrvar, A., & Batagelj, V. (2018). *Exploratory social network analysis with Pajek: Revised and expanded edition for updated software* (Vol. 46). Cambridge: Cambridge University Press.

Devi, J. C., & Poovammal, E. (2016). An analysis of overlapping

community detection algorithms in social networks. *Procedia Computer Science*, *89*, 349–358.

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv:1810.04805*, 1–16. Retrieved from https://doi.org/10.48550/arXiv.1810.04805

Dietrich, N., Kentheswaran, K., Ahmadi, A., Teychené, J., Bessière, Y., Alfenore, S., ... Hébrard, G. (2020). Attempts, successes, and failures of distance learning in the time of COVID-19. *Journal of Chemical Education*, *97*, 2448–2457.

Ding, Y., Chowdhury, G. G., & Foo, S. (2001). Bibliometric cartography of information retrieval research by using co-word analysis. *Information processing & management*, *37*(6), 817–842.

Douven, I., & Meijs, W. (2007). Measuring coherence. *Synthese*, *156*(3), 405–425.

Dulli, S., Polpettini, P., & Trotta, M. (2004). *Text mining: teoria e applicazioni*. Milan: Franco Angeli.

Eberendu, A. C. (2016). Unstructured Data: an overview of the data of Big Data. *International Journal of Computer Trends and Technology*, *38*(1), 46–50.

Egger, R., & Yu, J. (2022). A Topic Modeling Comparison Between LDA, NMF, Top2Vec, and BERTopic to Demystify Twitter Posts. *Frontiers in Sociology*, *7*, 1–16.

Fayyad, U., Piatetsky-Shapiro, G., & Smyth, P. (1996). From data mining to knowledge discovery in databases. *AI magazine*, *17*(3), 37–54.

Feldman, R., & Dagan, I. (1995). Knowledge Discovery in Textual Databases (KDT). In *Kdd* (Vol. 95, pp. 112–117).

Févotte, C., & Idier, J. (2011). Algorithms for nonnegative matrix

factorization with the β-divergence. *Neural computation*, *23*(9), 2421–2456.

Floridi, L. (2012). Big data and their epistemological challenge. *Philosophy & Technology*, *25*(4), 435–437.

Floridi, L. (2014). *The fourth revolution: How the infosphere is reshaping human reality*. Oxford: Oxford University Press.

Frawley, W. J., Piatetsky-Shapiro, G., & Matheus, C. J. (1992). Knowledge Discovery in Databases: An Overview. *AI magazine*, *13*(3), 57–70.

Freeman, L. C. (1977). A set of measures of centrality based on betweenness. *Sociometry*, 35–41.

Galdieri, M. (2020). Flessibilità e adattamento al cambiamento nella trasposizione didattica a distanza. *Education Sciences & Society-Open Access*, *11*, 477–503.

Galluccio, C., Crescenzi, F., & Petrucci, A. (2021). The italian newspapers' narrative on distance learning during the covid-19 pandemic. *IJAS*, 107-122.

Galluccio, G. (2021). The architect as a "semantic agent" in the dialogue between new practices and digital technologies. *TECHNE - Journal of Technology for Architecture and Environment*, 183–191.

Gandomi, A., & Haider, M. (2015). Beyond the hype: Big data concepts, methods, and analytics. *International Journal of Information Management*, *35*(2), 137–144.

Garz, M. (2012). Job insecurity perceptions and media coverage of labor market policy. *Journal of Labor Research*, *33*, 528–544.

Greene, D., & Cunningham, P. (2006). Practical solutions to the problem of diagonal dominance in kernel document clustering. In W. Cohen & A. Moore (Eds.), *ICML'06: Proceedings of the*

23rd international conference on Machine learning (pp. 377–384). New York: Association for Computing Machinery.

Greene, D., O'Callaghan, D., & Cunningham, P. (2014). How many topics? Stability analysis for topic models. In T. Calders, F. Esposito, E. Hüllermeier, & R. Meo (Eds.), *Ecml pkdd 2014: Machine learning and knowledge discovery in databases* (pp. 498–513). Springer.

Grefenstette, G., & Tapanainen, P. (1994). What is a word, what is a sentence?: problems of Tokenisation. *Rank Xerox Research Centre Meylan*, 1–9.

Griffiths, T. L., & Steyvers, M. (2004). Finding scientific topics. *Proceedings of the National academy of Sciences*, *101*, 5228–5235.

Grootendorst, M. (2022). BERTopic: Neural topic modeling with a class-based TF-IDF procedure. *arXiv:2203.05794*, 1–10. Retrieved from https://doi.org/10.48550/arXiv.2203.05794

Guangxu, X., Yaliang, L., Wayne Xin, Z., Jing, G., & Aidong, Z. (2017). A correlated topic model using word embeddings. In C. Sierra (Ed.), *IJCAI'17: Proceedings of the 26th International Joint Conference on Artificial Intelligence* (Vol. 17, pp. 4207–4213). Palo Alto: AAAI Press.

Hamm, A., & Odrowski, S. (2021). Term-community-based topic detection with variable resolution. *Information*, *12*(6), 221–252.

Hofmann, T. (1999). Probabilistic latent semantic indexing. In F. Gey, M. Hearst, & R. Tong (Eds.), *SIGIR'99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 50–57). New York: Association for Computing Machinery.

Hofmann, T. (2001). Unsupervised learning by probabilistic latent

semantic analysis. *Machine learning*, *42*(1), 177–196.

Hotho, A., Nürnberger, A., & Paaß, G. (2005). A brief survey of text mining. *Journal for Computational Linguistics and Language Technology*, *20*(1), 19–62.

Hubert, L., & Arabie, P. (1985). Comparing partitions. *Journal of classification*, *2*(1), 193–218.

Hull, D. A. (1996). Stemming algorithms: A case study for detailed evaluation. *Journal of the American Society for Information Science*, *47*(1), 70–84.

Iwata, T., Saito, K., Ueda, N., Stromsten, S., Griffiths, T., & Tenenbaum, J. (2007). Parametric Embedding for Class Visualization. *Neural Computation*, *19*(9), 2536–2556.

Kadhim, A. I. (2018). An Evaluation of Preprocessing Techniques for Text Classification. *International Journal of Computer Science and Information Security (IJCSIS)*, *16*(6), 22–32.

Kannan, S., & Gurusamy, V. (2014). Preprocessing Techniques for Text Mining. *International Journal of Computer Science & Communication Networks*, *5*(1), 7–16.

Kim, H., & Park, H. (2008). Nonnegative matrix factorization based on alternating nonnegativity constrained least squares and active set method. *Journal on matrix analysis and applications (SIAM)*, *30*(2), 713–730.

Kim, M., & Sayama, H. (2020). The power of communities: A text classification model with automated labeling process using network community detection. In N. Masuda, K. I. Goh, T. Jia, J. Yamanoi, & H. Sayama (Eds.), *Proceedings of NetSci-X 2020: Sixth International Winter School and Conference on Network Science* (pp. 231–243). New York City: Springer International Publishing.

Kleinberg, J. M., Kumar, R., Raghavan, P., Rajagopalan, S., & Tomkins, A. S. (1999). The web as a graph: Measurements, models, and methods. In T. Asano, H. Imai, D. T. Lee, S. Nakano, & T. Tokuyama (Eds.), *Computing and Combinatorics* (pp. 1–17). Heidelberg: Springer-Verlag.

Kodinariya, T. M., & Makwana, P. R. (2013). Review on determining number of Cluster in K-Means Clustering. *International Journal*, *1*(6), 90–95.

Kuzelj, M., & Šamija, K. (2020). Distance learning caused by the COVID-19 pandemic in Croatia: What do newspaper portals actually deliver to readers? In K. Marko et al. (Eds.), *2020 43rd International Convention on Information, Communication and Electronic Technology (MIPRO)* (pp. 854–859). Opatija: IEEE.

Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., & Soricut, R. (2019). Albert: A lite bert for self-supervised learning of language representations. *arXiv:1909.11942*, 1–17. Retrieved from https://doi.org/10.48550/arXiv.1909.11942

Lancichinetti, A., & Fortunato, S. (2009). Community detection algorithms: a comparative analysis. *Physical Review E*, *80*(5), 1–11.

Lancichinetti, A., Sirer, M. I., Wang, J. X., Acuna, D., Körding, K., & Amaral, L. A. N. (2015). High-reproducibility and high-accuracy method for automated topic classification. *Physical Review X*, *5*(1), 1–11.

Landauer, T. K., Foltz, P. W., & Laham, D. (1998). An introduction to latent semantic analysis. *Discourse processes*, *25*(2-3), 259–284.

Lang, K. (1995). Newsweeder: Learning to filter netnews. In A. Prieditis & S. Russell (Eds.), *Machine Learning: Proceedings of the Twelfth International Conference on Machine Learning covers the papers*

presented at the Twelfth International Conference on Machine Learning (ML95) (pp. 331–339). Amsterdam: Elsevier.

Lee, D. D., & Seung, H. S. (1999). Learning the parts of objects by non-negative matrix factorization. *Nature*, *401*(6755), 788–791.

Lee, D. D., & Seung, H. S. (2000). Algorithms for Non-negative Matrix Factorization. In T. Leen, T. Dietterich, & V. Tresp (Eds.), *Advances in Neural Information Processing Systems* (Vol. 13, pp. 1–7). Cambridge (Massachusetts): MIT Press.

Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C. H., & Kang, J. (2020). BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, *36*(4), 1234–1240.

Lee, Y., Lee, Y., Seong, J., Stanescu, A., & Hwang, C. S. (2020). A comparison of network clustering algorithms in keyword network analysis: a case study with geography conference presentations. *International Journal of Geospatial and Environmental Research*, *7*(3), 1–14.

Lin, C. J. (2007). Projected gradient methods for nonnegative matrix factorization. *Neural computation*, *19*(10), 2756–2779.

Lin, J., & Ban, Y. (2013). Complex network topology of transportation systems. *Transport reviews*, *33*(6), 658–685.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv:1907.11692*, 1–13. Retrieved from https://doi.org/10.48550/arXiv.1907.11692

Lucisano, P. (2020). Fare ricerca con gli insegnanti. I primi risultati dell'indagine nazionale SIRD "Per un confronto sulle modalità di didattica a distanza adottate nelle scuole italiane nel periodo di emergenza COVID-19". *Lifelong Lifewide Learning*, *16*, 3–25.

Manning, C., Raghavan, P., & Schutze, H. (2008). Term weighting, and the Vector Space Model. *Introduction to information retrieval*, 109–133.

Mansell, R. (2010). The life and times of the information society. *Prometheus*, *28*(2), 165–186.

Masucci, A. P., Kalampokis, A., Eguíluz, V. M., & Hernández-García, E. (2011). Wikipedia information flow analysis reveals the scale-free architecture of the semantic space. *PloS one*, *6*(2), 1–7.

Mayer-Schönberger, V., & Cukier, K. (2013). *Big data: A revolution that will transform how we live, work, and think*. Boston: Houghton Mifflin Harcourt.

McAfee, A., Brynjolfsson, E., Davenport, T. H., Patil, D. J., & Barton, D. (2012). Big data: the management revolution. *Harvard Business Review*, *90*(10), 60–68.

McCombs, M. E., & Shaw, D. L. (1972). The agenda-setting function of mass media. *Public opinion quarterly*, *36*, 176–187.

McInnes, L., Healy, J., & Melville, J. (2018). UMAP: Uniform manifold approximation and projection for dimension reduction. *arXiv:1802.03426*, 1–63. Retrieved from https://doi.org/10.48550/arXiv.1802.03426

Mehri, A., Darooneh, A. H., & Shariati, A. (2012). The complex networks approach for authorship attribution of books. *Physica A: Statistical Mechanics and its Applications*, *391*(7), 2429–2437.

Mothe, J., Mkhitaryan, K., & Haroutunian, M. (2017). Community detection: Comparison of state of the art algorithms. In S. Shoukourian (Ed.), *2017 Computer Science and Information Technologies (CSIT)* (pp. 125–129). Piscataway: IEEE.

Nayak, A. S., Kanive, A. P., Chandavekar, N., & Balasubramani, R.

(2016). Survey on Pre-processing Techniques for Text Mining. *International Journal of Engineering and Computer Science*, *5*(6), 16875–16879.

Negara, E. S., & Andryani, R. (2018). A review on overlapping and non-overlapping community detection algorithms for social network analytics. *Far East Journal of Electronics and Communications*, *18*(1), 1–27.

Newman, M. E. J. (2006). Finding community structure in networks using the eigenvectors of matrices. *Physical review E*, *74*(3), 1–22.

Newman, M. E. J. (2018). *Networks*. Oxford: Oxford university press.

O'Callaghan, D., Greene, D., Carthy, J., & Cunningham, P. (2015). An analysis of the coherence of descriptors in topic modeling. *Expert Systems with Applications*, *42*(13), 5645–5657.

Palareti, F. (2020). Didattica a distanza: strumenti e criticità. *Bibelot: notizie dalle biblioteche toscane*, *26*. Retrieved from https://riviste.aib.it/index.php/bibelot/article/view/12032

Palla, G., Derényi, I., Farkas, I., & Vicsek, T. (2005). Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, *435*(7043), 814–818.

Palmer, D. D. (2000). Tokenisation and Sentence Segmentation. *Handbook of Natural Language Processing*, 11–35.

Pang, B., & Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval*, *2*(1-2), 1–135.

Pennacchiotti, M., & Pantel, P. (2009). Entity extraction via ensemble semantics. In P. Koehn & R. Mihalcea (Eds.), *EMNLP'09: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing* (pp. 238–247). Stroudsburg: Association

for Computational Linguistics.

Raghavan, U. N., Albert, R., & Kumara, S. (2007). Near linear time algorithm to detect community structures in large-scale networks. *Physical Review E*, *76*(3), 1–11.

Ranganath, R., Gerrish, S., & Blei, D. (2014). Black Box Variational Inference. In S. Kaski & J. Corander (Eds.), *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics* (Vol. 33, pp. 814–822). Reykjavik: Proceedings of Machine Learning Research.

Rivoltella, P. C. (2020). Tecnologia più condivisione: così si può fare buon e-learning. *Milano: Avvenire.it*, *17*. Retrieved from https://www.avvenire.it/opinioni/pagine/tecnologia-pi-condivisione-cos-si-pu-fare-buon-elearning

Röder, M., Both, A., & Hinneburg, A. (2015). Exploring the space of topic coherence measures. In X. Cheng, H. Li, E. Gabrilovich, & J. Tang (Eds.), *WSDM'15: Proceedings of the Eighth ACM International Conference on Web Search and Data Mining* (pp. 399–408). New York: Association for Computing Machinery.

Roncaglia, G. (2020). *L'età della frammentazione: cultura del libro e scuola digitale*. Roma: Giusti Laterza & Figli Spa.

Roncaglia, G., Benigno, V., Caruso, G., Chifari, A., Ferlino, L., Fulantelli, G., ... Priore, M. (2021). L'educazione e la distanza: le risposte della scuola e il ruolo delle biblioteche scolastiche. *Biblioteche oggi Trends*, *6*, 110–134.

Rosner, F., Hinneburg, A., Röder, M., Nettling, M., & Both, A. (2014). Evaluating topic coherence measures. *arXiv:1403.6397*. Retrieved from https://doi.org/10.48550/arXiv.1403.6397

Sabidussi, G. (1966). The centrality index of a graph. *Psychometrika*, *31*(4), 581–603.

Saif, H., Fernández, M., He, Y., & Alani, H. (2014). On Stopwords, Filtering and Data Sparsity for Sentiment Analysis of Twitter. In N. Calzolari et al. (Eds.), *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)* (pp. 810–817). Reykjavik: European Language Resources Association (ELRA).

Salerno, M. D., Tataru, C. A., & Mallory, M. R. (2015). *Word Community Allocation: Discovering Latent Topics via Word Co-Occurrence Network Structure.* Retrieved from http://snap.stanford.edu/class/cs224w-2015/projects2015/WordCommunityAllocation.pdf

Salton, G., & McGill, M. J. (1983). *Introduction to modern information retrieval.* New York: McGraw-Hill.

Sánchez-Franco, M. J., & Rey-Moreno, M. (2022). Do travelers' reviews depend on the destination? An analysis in coastal and urban peer-to-peer lodgings. *Psychology & Marketing*, *39*(2), 441–459.

Sayyadi, H., & Raschid, L. (2013). A graph analytical approach for topic detection. *ACM Transactions on Internet Technology (TOIT)*, *13*(2), 1–23.

Schnegg, M., & Bernard, H. R. (1996). Words as actors: A method for doing semantic network analysis. *CAM Journal*, *8*(2), 7–10.

Schütze, H., Manning, C. D., & Raghavan, P. (2008). *Introduction to information retrieval* (Vol. 39). Cambridge: Cambridge University Press.

Song, K., Tan, X., Qin, T., Lu, J., & Liu, T. Y. (2020). Mpnet: Masked and permuted pre-training for language understanding. *Advances in Neural Information Processing Systems*, *33*, 16857–16867.

Srividhya, V., & Anitha, R. (2010). Evaluating Preprocessing Techniques in Text Categorization. *International journal of computer*

science and application, *47*(11), 49–51.

Stevens, K., Kegelmeyer, P., Andrzejewski, D., & Buttler, D. (2012). Exploring topic coherence over many models and many topics. In J. Tsujii, J. Henderson, & M. Paşca (Eds.), *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning.* (pp. 952–961). Jeju Island: Association for Computational Linguistics.

Steyvers, M., & Griffiths, T. (2007). Probabilistic Topic Models. In *Handbook of Latent Semantic Analysis* (pp. 439–460). London: Psychology Press.

Tang, D., Qin, B., & Liu, T. (2015). Document modeling with gated recurrent neural network for sentiment classification. In L. Màrquez, C. Callison-Burch, & J. Su (Eds.), *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* (pp. 1422–1432). Lisbon: Association for Computational Linguistics.

Tausch, F., & Zumbuehl, M. (2018). Stability of risk attitudes and media coverage of economic news. *Journal of Economic Behavior & Organization*, *150*, 295–310.

Traag, V. A., Waltman, L., & Van Eck, N. J. (2019). From louvain to leiden: guaranteeing well-connected communities. *Scientific Reports*, *9*(1), 1–12.

Tripathi, B., Parthasarathy, S., Sinha, H., Raman, K., & Ravindran, B. (2019). Adapting community detection algorithms for disease module identification in heterogeneous biological networks. *Frontiers in Genetics*, *10*, 1–17.

Usai, A., Pironti, M., Mital, M., & Mejri, C. A. (2018). Knowledge discovery out of text data: a systematic review via text mining.

Journal of Knowledge Management, *22*, 1471–1488.

Uysal, A. K., & Gunal, S. (2014). The impact of preprocessing on text classification. *Information processing & management*, *50*(1), 104–112.

Valente, T. W., Coronges, K., Lakon, C., & Costenbader, E. (2008). How correlated are network centrality measures? *Connection*, *28*(1), 16–26.

Van Dijk, J. A. G. M. (2020). *The network society*. London: SAGE publications.

Wallach, H. M., Murray, I., Salakhutdinov, R., & Mimno, D. (2009). Evaluation methods for topic models. In A. Danyluk, L. Bottou, & M. Littman (Eds.), *ICML'09: Proceedings of the 26th Annual International Conference on Machine Learning* (pp. 1105–1112). New York: Association for Computing Machinery.

Wang, Y., Sabzmeydani, P., & Mori, G. (2007). Semi-latent dirichlet allocation: A hierarchical model for human action recognition. In A. Elgammal, B. Rosenhahn, & R. Klette (Eds.), *Proceedings of the 2nd Conference on Human Motion: Understanding, Modeling, Capture and Animation* (Vol. 4814, pp. 240–254). Heidelberg: Springer-Verlag.

Wasserman, S., & Faust, K. (1994). *Social network analysis: Methods and applications*. Cambridge: Cambridge University Press.

Webster, J. J., & Kit, C. (1992). Tokenization as the initial phase in NLP. In A. Zampolli (Ed.), *COLING'92: Proceedings of the 14th conference on Computational linguistics* (Vol. 4, pp. 1106–1110). Stroudsburg: Association for Computational Linguistics.

Weiss, S. M., Indurkhya, N., Zhang, T., & Damerau, F. J. (2005). From textual information to numerical vectors. In *Text Mining* (pp. 15–46). Heidelberg: Springer-Verlag.

Xie, J., Kelley, S., & Szymanski, B. K. (2013). Overlapping Community Detection in Networks: The State-of-the-Art and Comparative Study. *ACM Computing Surveys*, *45*(4), 1–35.

Xie, J., Szymanski, B. K., & Liu, X. (2011). SLPA: Uncovering Overlapping Communities in Social Networks via a Speaker-Listener Interaction Dynamic Process. In M. Spiliopoulou et al. (Eds.), *2011 IEEE 11th International Conference on Data Mining Workshops (ICDM 2011)* (pp. 344–349). Piscataway: IEEE Computer Society.

Yao, Y., Vehtari, A., Simpson, D., & Gelman, A. (2018). Yes, but Did It Work?: Evaluating Variational Inference. In J. G. Dy & A. Krause (Eds.), *Proceedings of the 35th International Conference on Machine Learning* (Vol. 80, pp. 5581–5590). Stockholm: Proceedings of Machine Learning Research.

Zihan, Z., Meng, F., Ling, C., & Mohammad-Reza, N. R. (2022). Is Neural Topic Modelling Better than Clustering? An Empirical Study on Clustering with Contextual Embeddings for Topics. *arXiv:2204.09874*, 1–8. Retrieved from https://doi.org/10.48550/arXiv.2204.09874