UNIVERSIDADE DE SÃO PAULO
Instituto de Ciências Matemáticas e de Computação

Employing syntactical dependency and a mesoscopic scale to model books' narratives through recurrence networks

Bárbara Côrtes e Souza

Dissertação de Mestrado do Programa de Pós-Graduação em Ciências de Computação e Matemática Computacional (PPG-CCMC)

ICMC USP
SÃO CARLOS

Bárbara Côrtes e Souza

Employing syntactical dependency and a mesoscopic scale to model books' narratives through recurrence networks

Dissertation submitted to the Instituto de Ciências Matemáticas e de Computação – ICMC-USP – in accordance with the requirements of the Computer and Mathematical Sciences Graduate Program, for the degree of Master in Science. *EXAMINATION BOARD PRESENTATION COPY*

Concentration Area: Computer Science and Computational Mathematics

Advisor: Prof. Dr. Diego Raphael Amancio

USP – São Carlos
August 2023

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados inseridos pelo(a) autor(a)

C229e

Côrtes e Souza, Bárbara
Employing syntactical dependency and a mesoscopic scale to model books' narratives through recurrence networks / Bárbara Côrtes e Souza; orientador Diego Raphael Amancio. -- São Carlos, 2023.
82 p.

Dissertação (Mestrado - Programa de Pós-Graduação em Ciências de Computação e Matemática Computacional) -- Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, 2023.

1. Processamento de Linguagem Natural. 2. Redes Complexas. 3. Escala mesoscópica. 4. Dependência sintática. 5. Redes de recorrência. I. Amancio, Diego Raphael, orient. II. Título.

Bibliotecários responsáveis pela estrutura de catalogação da publicação de acordo com a AACR2:
Gláucia Maria Saia Cristianini - CRB - 8/4938
Juliana de Souza Moraes - CRB - 8/6176

Bárbara Côrtes e Souza

Aplicando dependência sintática e uma escala mesoscópica para modelar narrativas de livros a partir de redes de recorrência

Dissertação apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP, como parte dos requisitos para obtenção do título de Mestra em Ciências – Ciências de Computação e Matemática Computacional. *EXEMPLAR DE DEFESA*

Área de Concentração: Ciências de Computação e Matemática Computacional

Orientador: Prof. Dr. Diego Raphael Amancio

USP – São Carlos
Agosto de 2023

ACKNOWLEDGEMENTS

Agradeço imensamente a todos que participaram de todo o meu processo até chegar no dia de hoje. Seja me apoiando, ouvindo meus desabafos, me incentivando e mil e uma outras coisas. Saibam que cada um de vocês foi muito importante para mim.

Agradeço à minha família por todo o apoio. Aos meus pais, Káthia e Sérgio, que sempre acreditaram em mim e me deram todo o suporte que eu precisei. Ao meu irmão, Bernardo, que sempre esteve ao meu lado inclusive durante a revisão do meu texto de qualificação.

Aos meus amigos, especialmente dos grupos "Universiotários" e "Acadêmicos". Foi muito importante para mim poder contar com vocês por todos esses anos. Vocês me inspiraram, motivaram e me ensinaram muito. Serei eternamente grata por cada um de vocês. E aos colegas pesquisadores que já passaram ou que ainda vão passar por este momento, força amigos!

Especialmente, à minha melhor amiga e namorada, Beatriz Monteiro. Obrigada por ter estado comigo em tantos momentos, e por tantos outros ainda por vir. E obrigada por sempre acreditar em mim, mesmo quando eu não acreditei nem por um segundo nesta sua leitura light de terça-feira.

Aos meus gatos, Loki e Luna, por todo o amor incondicional. Por estarem ao meu lado durante todo o tempo em que li, implementei, escrevi e pesquisei cada etapa deste trabalho.

Aos meus amigos e companheiros de pós graduação, Giovana Daniele e Leo Paschoal. Obrigada por tornarem minha trajetória no mestrado muitíssimo mais agradável amigos.

Ao meu orientador Dr. Diego Raphael Amancio por toda a nossa história, que começou em 2019 na disciplina de Compiladores. Obrigada por todos os ensinamentos, apoio, carinho e liberdade, e por todo o trabalho que fizemos juntos.

Aos meus colegas do IFSC e toda a sua colaboração para a conclusão deste trabalho, Filipi, Henrique e Prof. Luciano da F. Costa.

À CAPES pelo apoio financeiro e pelo interesse neste estudo. Ao ICMC e à USP por toda a experiência incrível que me proporcionaram nestes últimos quase 8 anos.

This research was supported by CAPES. The opinions, assumptions, conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of CAPES.

RESUMO

SOUZA, B. C. **Aplicando dependência sintática e uma escala mesoscópica para modelar narrativas de livros a partir de redes de recorrência**. 2023. 82 p. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2023.

Nos últimos anos, a ciência tem sido fortemente influenciada pelo contínuo aumento no volume de informações disponíveis à pesquisa. Especificamente, o crescimento da quantidade de dados textuais desempenhou um papel fundamental no desenvolvimento e na apresentação de novas metodologias para abordar desafios na área de processamento de textos. Diversas abordagens inovadoras têm surgido, com enfoque em diferentes componentes da linguística, como léxico, sintaxe e semântica. O Processamento de Linguagem Natural, por exemplo, é um campo multidisciplinar que aborda a interação entre linguagens naturais e computadores. Alguns exemplos de problemas dessa área são: detecção de tópicos, classificação de textos, estilometria, sumarização automática, entre outros. Dado que linguagens naturais são consideradas sistemas complexos, é apropriado que sejam representadas por redes complexas, para auxiliar na resolução desses diferentes tipos de problemas. Um conhecido método de modelagem de textos é a rede de adjacência de palavras, na qual cada nó mapeia uma palavra do texto e arestas são criadas entre termos que ocorrem em sequência no texto. Neste projeto de Mestrado, no entanto, o foco é em uma escala mesoscópica mais abrangente, visando a capturar o contexto geral da narrativa. Nessa metodologia, um nó se refere a uma sequência de parágrafos do texto, e arestas são criadas entre os mais similares. Adicionalmente, uma análise de dependência sintática é aplicada para aumentar o nível de *informatividade* e, portanto, obter uma performance superior em capturar o contexto semântico de um texto. Finalmente, é possível extrair medidas de rede significativas para sua caracterização, incluindo acessibilidade, simetria e a proposta Assinatura de Recorrência, como forma de capturar as propriedades topológicas que refletem o contexto narrativo. Diversas validações de método foram executadas, incluindo uma comparação com outras medidas de rede triviais, dois experimentos para diferenciar entre textos reais e randomizados e entre diferentes gêneros literários, e, finalmente, uma comparação do método proposto com outras abordagens mais ortodoxas na literatura: redes de co-ocorrência e doc2vec.

Palavras-chave: Redes Complexas, Escala mesoscópica, Dependência sintática, PLN, Redes de recorrência.

ABSTRACT

SOUZA, B. C. **Employing syntactical dependency and a mesoscopic scale to model books' narratives through recurrence networks**. 2023. 82 p. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2023.

In recent years, science has been deeply impacted by the growing amount of data available for research. Specifically, the continuous increase of textual data availability has been essential for the development and proposal of new methodologies to tackle text processing problems. There are several new approaches that focus on different components of linguistics, such as lexicon, syntax and semantics. Natural Language Processing, for example, is a multidisciplinary field that concerns the interaction between natural languages and computers. Some examples of problems in this field are: topic detection, text classification, stylometry, automatic summarization, and others. Since natural languages are actually complex systems, it is also appropriate to represent them as complex networks, to help address these various challenges. One well known example of text modelling method is the word adjacency network, that maps each of the words in a text into nodes, and create an edge between any pair of terms that occur adjacent to each other in the text. In this Masters' work, however, we focus on a larger, mesoscopic scale with the intent of capturing the overall context of a narrative. In this methodology, a single node refers to a sequence of paragraphs in the text, and the edges are created between the most similar ones. Additionally, we apply syntactical dependency knowledge to increase informativeness and, therefore, obtain a better performance on grasping the contextual semantics of the text. Finally, one can extract significant network measures in order to characterize it, including accessibility, symmetry and the new proposed recurrence signature, as a manner of capturing topological properties that reflect the narrative's context. Several method validations have been performed, including a comparison with other trivial measures, two experiments to discriminate real from meaningless texts and between literary genres and, finally, a comparison of the current method to other orthodox approaches, namely co-occurrence networks and doc2vec.

Keywords: Complex Networks, Mesoscopic scale, Syntactical dependency, NLP, Recurrence networks.

LIST OF FIGURES

LIST OF ALGORITHMS

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

BA	Barabási-Albert
BOW	Bag-Of-Words
CART	Classification and Regression Tree
ER	Erdös-Rényi
FR	Fruchterman-Reingold
KNN	K-Nearest Neighbors
NLP	Natural Language Processing
TF-IDF	Term Frequency-Inverse Document Frequency
WS	Watts-Strogatz
WWW	World Wide Web

LIST OF SYMBOLS

G — Graph

V — Set of vertices of a graph

E — Set of edges of a graph

M — Number of edges in a graph

A — Adjacency matrix of a graph

w — Edge weight

N — Number of nodes in a graph

K — Degree

K^{in} — In-degree

K^{out} — Out-degree

s — Strength

s^{in} — In-strength

s_i^{out} — Out-strength

h — Concentric level

α — Accessibility

Sb — Backbone symmetry

Sm — Merged symmetry

S — Concentric symmetry

cc — Clustering coefficient

N_Δ — Number of triangles in a graph

N_3 — Number of triples in a graph

w — Word

d — Document

D — Set of documents

tf-idf — Term frequency-inverse document frequency

tf — term frequency

idf — inverse document frequency

f_w^d — Frequency of a word w in a document d

f_w^D — Total frequency of the word w in the set of documents D

N_w — Number of documents that contain word w

sim — Cosine similarity

θ — Angle

O — Organized text

p — Paragraph

w — Word

P — Paragraph window

Δ — Paragraph window size

T — Threshold for the network's average degree

RS — Recurrence signature

ρ — Pearson correlation

r_s — Spearman correlation

KS — Kolmogorov-Smirnov

CONTENTS

CHAPTER

1

INTRODUCTION

1.1 Context and motivation

In the age of information, science has been deeply empowered by the continuous increase of data availability and the associated improvement of the ability to discover, mine and process it. Many sorts of data have participated in this information expansion, collaborating for the development of various study areas, from the exact sciences to humanities. In this Master's work, however, we focus on the textual data and its applications to tackle real-world problems, and how this growing amount of studied data has collaborated to the proposition of new methodologies.

Traditionally, text processing techniques were based only on straightforward statistical metrics, mostly involving word frequency (ALTMANN; PIERREHUMBERT; MOTTER, 2009). However, with the demand for tackling more complex problems, more sophisticated approaches gained a lot of space in researches. Natural languages are composed of several different components (BRISCOE, 2014), including morphology, lexicon, syntax and semantics. Therefore, there is plenty of room for strategies that extrapolate the text vocabulary, including studying the elements of a text, their importance, how they relate to each other (syntax) and how it all contributes to the construction of the meaning behind that text (semantics).

Natural Language Processing (NLP) is a multidisciplinary area of study concerned with the interaction between computers and natural languages, thus, encompassing all the previously mentioned text processing problems. With the proposition of different techniques (such as *Bag-Of-Words* (BUNT; BOS; PULMAN, 2013), *word embeddings* (MIKOLOV *et al.*, 2013a) and *co-occurrence networks* (BREDE; NEWTH, 2008)), this field aims to represent specific language properties in a way that they can be interpreted and elaborated automatically by a machine. Subsequently, the greater complexity of the methods considered allows various complex tasks to be tackled, such as topic detection (ALSUMAIT; BARBARÁ; DOMENICONI, 2008), story flow representation (LIU *et al.*, 2013) and many others, that will be discussed later in this document.

Since natural languages can be understood as complex systems (STEELS, 2002), it is possible to affirm that the modelling of texts through complex networks is appropriate. Complex networks are graphs used to represent real-life systems, where each node is associated to an element from the system and the edges indicate an existing relationship between the connected elements. There are many other examples of real systems that can be modelled with networks, including the Internet, the World Wide Web, the telephone grid, social networks, and others, all mentioned in (NEWMAN, 2010). And there are also several examples of complex network approaches to represent texts and solve NLP problems, which is the main area of study comprising the research proposed in this Masters' work.

A well known and relatively simple method of modelling texts using complex networks is the *word adjacency network*, also referred to as *co-occurrence network*. In summary, each node represents a word of the text, and two nodes are connected if, and only if, they occur adjacent to each other in the text at least once. When combined with common NLP techniques that will be discussed hereafter, these approaches have been successfully applied to tackle various problems, such as authorship recognition (AMANCIO, 2015b), automatic translation (AMANCIO *et al.*, 2012), sentiment analysis (ESTRADA, 2011), stylometry (AMANCIO, 2015a), text classification (ARRUDA; COSTA; AMANCIO, 2016), among others.

However, while considering co-occurrence networks, the mentioned approaches focus only on a microscopic scale of the texts, which makes it hard to retrieve any upper level notion of the text semantic complexity. To solve this issue, in this Master's work, we employ a mesoscopic scale to propose a new method that focuses on larger elements of the text, namely, a sequence of paragraphs. By following a similar methodology, the authors in (ARRUDA *et al.*, 2016) were able to grasp the semantic context of a book's narrative, which is also the main goal of this research. Additionally, syntactical dependency analysis is incorporated into the study as linguistic knowledge to collaborate for an increase of informativeness (LEITE *et al.*, 2007).

In order to further tackle this objective, it is proposed a new modelling based on recurrence networks (DONNER *et al.*, 2010). In texts, there are short and long range relationships that are constructed in the narrative. For example, characters, locations and actions, that can be identified through their syntactical role in the text, will recur along the text while the story is built. Then, these structures will, by the co-occurrence of words for example, create long range links throughout the text, a pattern that can be modelled through recurrence networks. Finally, it will be possible to capture the mesoscopic relationships between the elements in the texts and, ultimately, to study and characterize them.

After establishing a recurrence network for a given book, it is necessary to employ strategies that will help translate the graph structure into quantitative or qualitative measures, that can later be used to assess different real world problems. For that, the usage of two topological and multi-scale measures, accessibility and symmetry, are introduced. Moreover, a new measure is idealized and proposed as a means to qualify the narrative's structure and unfolding throughout

a text, considering its semantic context: the recurrence signature. These measures and how well they work in assessing the different sorts of inquiries discussed in this Masters' research will be addressed in detail later on this document.

Finally, the following sections will present, respectively, a summary of the objectives of this Master's work and how the document was organized to explain some important background concepts, the methodology proposed and, finally, the results obtained.

1.2 Objectives

For this Master's research, the main objective is to *study text narratives and their semantic meaning from a mesoscopic perspective by taking into account their syntactical dependencies and representing them as recurrence networks (DONNER et al., 2010)*. By text narratives and their semantic meaning, we mean the actual story built and described along the text: the characters involved, their actions and its developments throughout time. For that, we consider a mesoscopic perspective - between the microscopic and the macroscopic - to analyze texts and represent them as recurrence networks, which allows us to study the elements and their relationships from a medium scale. This approach enables one to see the short and/or long range relationships between elements in the text (e.g. word adjacency and content similarity). These relationships usually indicate the occurrence of recurrent situations in time or space along the story, for example, a character that recurs along the text will create long range links in the recurrence network.

The motivation for this modelling approach is based on the fact that there are currently not many network-based methods in the literature that admit a larger scale to try and capture the contextual meaning of a narrative. The mesoscopic scale is definitive to overcome the limitations of micro-scale approaches (e.g. word adjacency), such as the fact that single words are typically not enough to fully define semantic context. By considering larger portions of the text (e.g. paragraphs), we can visualize the correspondence of contexts in distant portions of the text, hence establishing long range connections. Therefore, we can provide valuable insights on how complex networks modelling of texts can be sophisticated enough to touch on the semantics of the language, which is still a challenge to this day.

With this generic objective in mind and the established definitions, we also specify the following specific goals for this Master's work.

- Propose a network modelling technique based on recurrence that is able to capture the narrative context of a book, that is, the generated network must be able to represent the semantics of what the story told in the book.
- Extract a set of measures from the network that is sufficient to capture and represent its individual properties, in a way that they can be used in different tasks, such as clusterization, property prediction, etc.

- Demonstrate the validity of our method by discriminating real texts from meaningless ones, that is, texts that actually tell a story from others that do not have any semantic meaning (e.g. randomized texts).
- Apply our technique to differentiate books and predict specific properties (e.g. literary genre).

1.3 Monograph organization

This document is organized as follows:

- Chapter 2: Explanation of the basic concepts about complex networks: some famous models, important measurements and how to represent texts as networks.
- Chapter 3: Brief discussion of some basic Natural Language Processing concepts such as pre-processing, *Bag of Words*, *TF-IDF* and syntactic analysis.
- Chapter 4: Discussion of a few related works that represent texts as networks and that study the story flow within the texts.
- Chapter 5: Explanation of the methodology proposed, since the pre-processing of the text until extracting new measures from the constructed networks.
- Chapter 6: Discussion of the experiments performed and the results achieved.
- Chapter 7: Summary of the contributions, limitations and future work of this Master's research.

CHAPTER

2

COMPLEX NETWORKS

In this chapter, we will introduce the Complex Networks area of study. We will describe some of its basic concepts and definitions, and some important network models and measurements. At last, we will provide an overview of how texts can be modelled as networks.

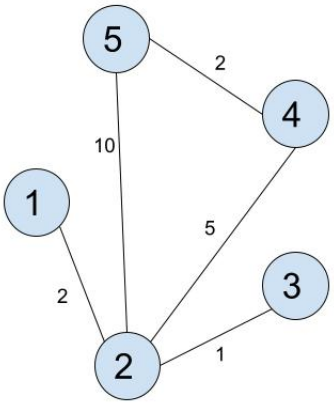
2.1 Introduction

Before understanding the definition of a complex network, we need to explain the definition of graphs or, simply, networks. According to (BOCCALETTI *et al.*, 2006), a graph G is defined by $G = (V,E)$, where $V = \{v_1, v_2, ...v_N\}$ is a set of the network vertices (also referred to as nodes or points) and $E = \{e_1, e_2, ...e_M\}$ is the set of edges (also referred to as links or connections), where N and M are the number of elements in V and E, respectively. An edge e_{ij} is defined by a pair of integers (i, j), indicating that the vertices i and j are connected by that edge. These pairs of elements can either be ordered or unordered, for the former case, the graph is considered directed, and an edge e_{ij} means that the source of the link is i, whereas j is the target. For the latter, the network is called undirected, and the edges e_{ij} and e_{ji} are actually the same.

A common form of representation for a graph is an adjacency matrix A . In this NxN matrix, each element A_{ij} represents a connection from a node i to a node j, meaning $A_{ij} = 1$ if the link between them exists, and $A_{ij} = 0$ otherwise. Another possibility is that the postition (i, j) of the matrix can be filled with any number $w \in \mathbb{R}$ representing the edge weight, in which case each edge e is now defined by a triple (i, j, w). Figure 1 shows an example of a simple weighted network and its adjacency matrix, which is symmetrical because the graph is undirected.

When talking about complex networks, we are referring to a specific set of graphs, with some common properties that make them more interesting to study. According to (NEWMAN, 2010), complex networks are representations of real-life systems, such as the Internet, the World

Figure 1 – Example of a weighted undirected network and its adjacency matrix.



Adjacency Matrix

	1	2	3	4	5
1		2			
2	2		1	5	10
3		1			
4		5			2
5		10		2	

Source: Elaborated by the author.

Wide Web (WWW), social networks, and others. When turning a large, complex system into this abstract representation, we capture only the patterns of the connections between the elements of that system, and, therefore, we are able to study and analyze its different properties, realizing how they actually represent the bigger system.

2.2 Network models

Even though complex networks are, usually, forms of representation of real world systems, it is also important to be able to generate a specific graph while having in mind a specific set of properties. Therefore, there are many different proposed network formation models and, in this section, we are going to explain some of the most famous ones.

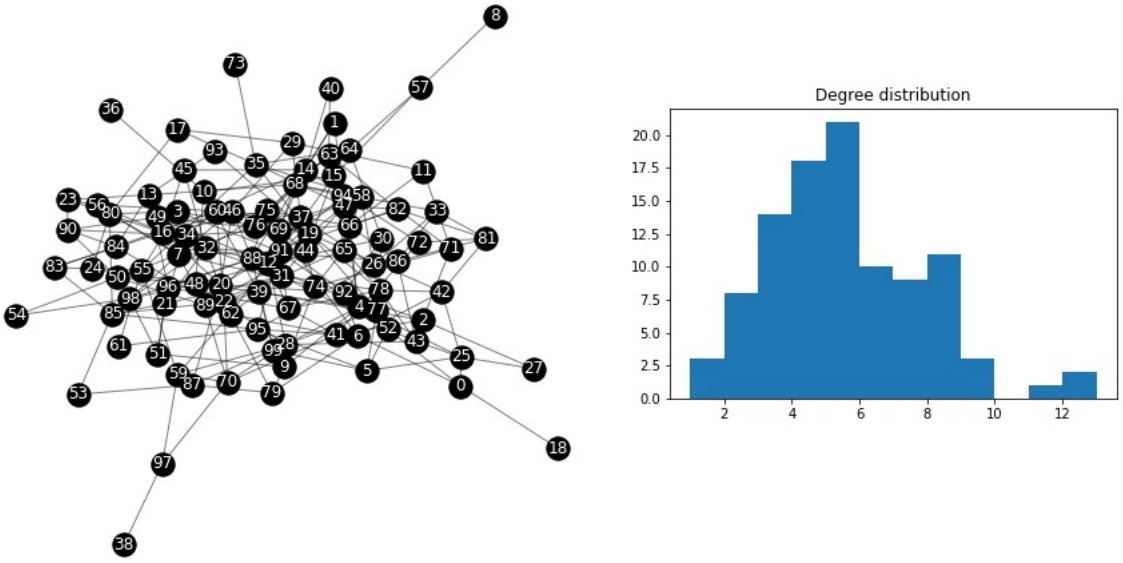
2.2.1 Erdös-Rényi model

The Erdös-Rényi (ER) model, defined in (ERDÖS; RÉNYI, 1959), is a well known method to generate a random network. Firstly, a graph is created with N vertices and, for every possible pair of nodes, an edge is added between them with a probability of p. The degree of the generated graph follows a binomial distribution and, for larger values of N, it tends to a Poisson. In Figure 2, we see an example of an ER random network and its degree distribution.

2.2.2 Watts-Strogatz model

The Watts-Strogatz (WS) model, proposed in (WATTS; STROGATZ, 1998) and also known as *small-world model*, is another important method for generating random graphs. It starts with a graph with N vertices, where each of them is connected to its k nearest neighbors on ring topology. Afterwards, each edge of the graph might be rewired (that is, to replace an edge (i, j) with another (i,k), where $i, j, k \in V$) with a probability of p. This model is known

Figure 2 – Example of network generated using the Erdös-Rényi model with $N = 100$ and $p = 0.5$ and its degree distribution. Generated using the *networkx* and *matplotlib* python libraries.



Source: Elaborated by the author.

for its small-world effect, which means that the topological distance between the vertices of the network is small. This feature is really important because it is widely seen in real world systems, such as the WWW, social networks, neuron networks, and others. See Figure 3 for two examples of WS networks.

2.2.3 Barabási-Albert model

Finally, we present the Barabási-Albert (BA) model, established in (BARABASI; ALBERT, 1999). This model's main characteristic is that the degree distribution of the network follows a power-law function, defined as:

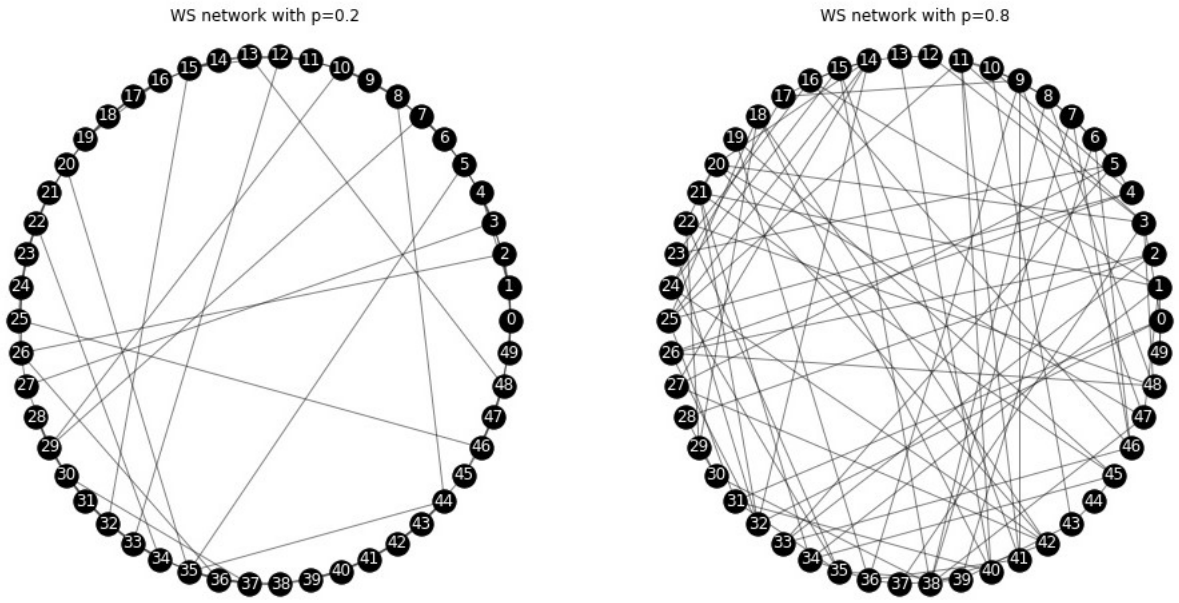
$$P(k) = k^{-\lambda},$$

where k is a degree value and λ is a constant. This distribution is also very common in real life systems, where we can observe the existence of *hubs* (highly connected nodes that are responsible for a significant fraction of the total number of edges in the network), which can also be observed in the WWW and many kinds of social networks.

To generate a BA network, we have to consider the following two rules:

- Growth: the process of increasing the number of nodes in the network; and
- Preferential attachment: a new node added to the network is more likely to link with the more connected nodes.

Figure 3 – Example of two networks generated using the Watts-Strogatz model with $N = 50$, $k = 4$ and the left one with $p = 0.2$ and the right one with $p = 0.8$. Generated using the *networkx* and *matplotlib* python libraries.



Source: Elaborated by the author.

Starting with a connected graph with m_0 nodes, at every step, we add a new vertex with m edges, where $m < m_0$, linking to the vertices already present in the network. The probability Π of this new node connecting with a specific node i is proportional to its degree k_i, and it is defined by:

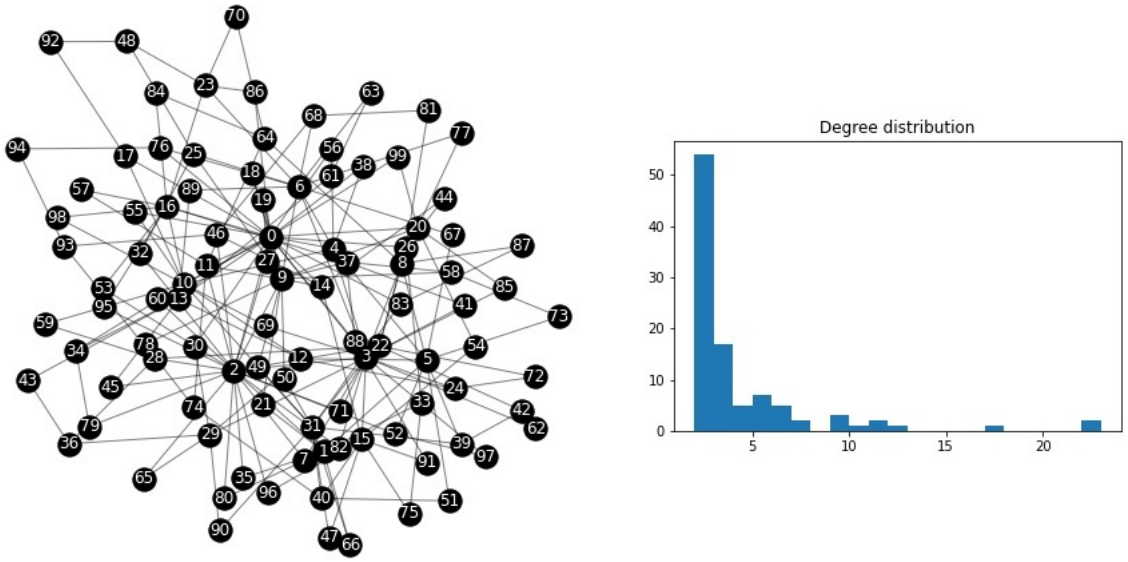
$$\Pi(k_i) = \frac{k_i}{\sum_j k_j}.$$

This process stops when the pre-defined final number of nodes N is reached. In Figure 4 we see an example of a BA network and its respective degree distribution.

2.3 Network bipartivity

According to (ESTRADA, 2011), a network can be called bipartite if all of its nodes can be split into two disjoint and non-empty sets, in a way that every edge of the network has both endpoints in nodes from different sets. That is: $V_1 \subset V$, $V_2 \subset V$, $V_1 \cup V_2 = V$ and an edge e_{ij} can exist only if $i \in V_1$ and $j \in V_2$ or vice versa. Bipartite networks have many important properties, such as being 2-colourable and not having cycles with odd length, which makes them an important network category for studies.

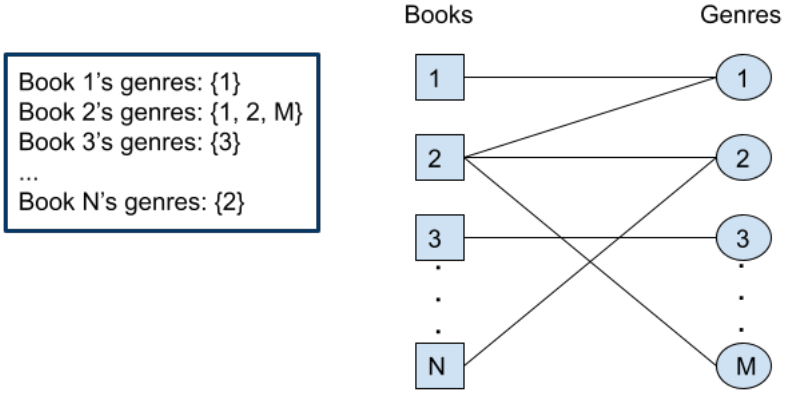
Figure 4 – Example of a network generated using the Barabasi-Albert model with $N = 100$ and $m = 2$ and its degree distribution. Generated using the *networkx* and *matplotlib* python libraries.



Source: Elaborated by the author.

Moreover, there are several real life systems that can be represented by bipartite networks, such as: exclusively heterosexual relationship between women (V_1) and men (V_2), commercial transactions between customers (V_1) and products (V_2), and citation networks between authors (V_1) and papers (V_2). As a more specific example, given a scenario where one book can have multiple literary genres assigned to it, one can also model this system as a bipartite network between books and genres, which is illustrated in Figure 5. This case study will be discussed further later on this document.

Figure 5 – Illustration of a bipartite network between books and their respective genres.



Source: Elaborated by the author.

2.4 Network measures

To characterize, identify and differentiate complex networks, we need to be able to extract numeric features from them. For that, there are many different proposed measurements for various applications, that represent and summarize numerous topological properties of the network and its elements. In this section, we will explain a few of these measures that will be important for the development of this Master's work.

2.4.1 Degree and strength

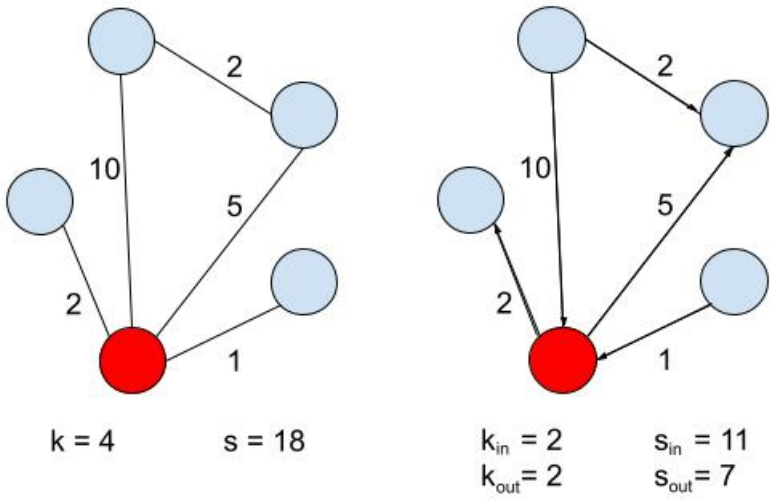
First, we will define a node's degree, the most straightforward and one of the most common network measurements, and from which derive a great number of other measures. A vertex's degree K_i is the number of edges connected to that node. For directed networks, this measurement is split into two: K_i^{in} and K_i^{out}, considering only the edges in which i is the target, for the former, and the source, for the former. See Figure 6 for an example of a node's degree, in both directed and undirected networks.

One of the measurements derived by the degree is a node's strength s_i. For any vertex i, s is defined as the sum of the weights of all the edges connected to i. For a weighted, undirected graph, it can be calculated as:

$$S_i = \sum_j w_{ij}.$$

Just like the degree measurement, a node's strength can also be defined for directed networks with s_i^{in} and s_i^{out}. In Figure 6, we also observe an example of a node's strength for both types of networks.

Figure 6 – Examples of degree and strength calculations for the red nodes. In the left, an undirected graph and red node's degree is 4 and its strength is 18. In the right, a directed graph: node's in-degree is 2, out-degree is 2, in-strength is 11 and out-strength is 7.

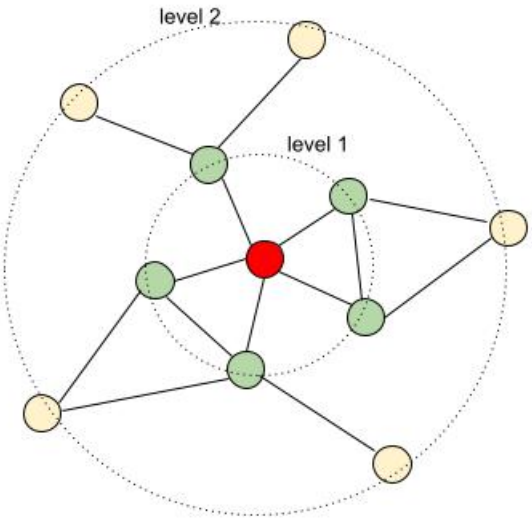


Source: Elaborated by the author.

2.4.2 Accessibility and symmetry

To define accessibility, first we need to explain another example of measurement derived by the notion of degree: the concept of concentric levels (COSTA; ANDRADE, 2007). A concentric level h of a node i is composed by all of the nodes in the graph that have a minimum distance to i equal to h. In Figure 7, we see an example of the concentric levels $h = 1$ and $h = 2$ for the red node.

Figure 7 – Examples of concentric levels for the red node. The nodes in green belong to level 1, while nodes in yellow belong to level 2.



Source: Elaborated by the author.

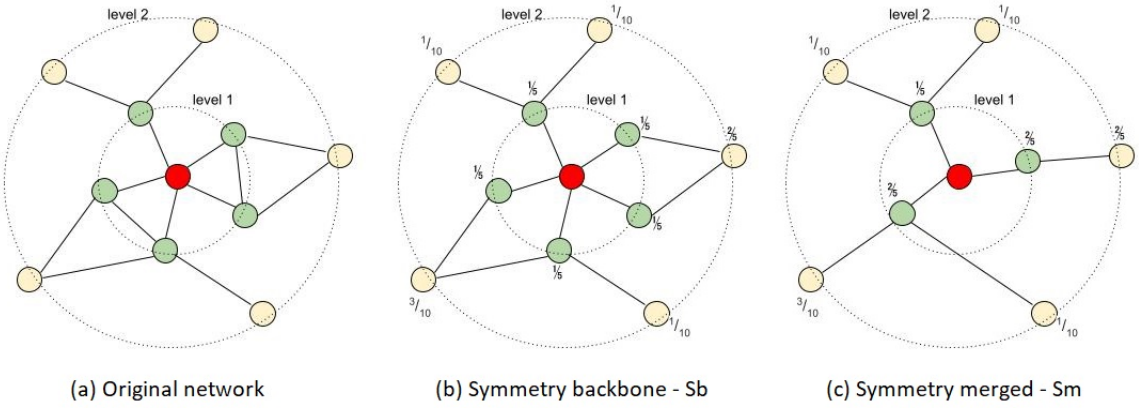
The accessibility (COSTA; ANDRADE, 2007) of a node is computed based on random walks along the graph (when a walker iteratively and randomly goes from one node to another through an edge). In one specific type of random walk (self-avoiding), the walker cannot go back to a node it has already been in. Based in this concept, one can define the accessibility α of a node i for a concentric level h as:

$$\alpha_i^{(h)} = \exp\left(-\sum_j (p_{ij}^{(h)} \times \log p_{ij}^{(h)})\right),$$

where $p_{ij}^{(h)}$ is the probability of the walker to reach the node j, h steps after leaving from i.

By deriving the concepts of concentric levels and accessibility, (SILVA *et al.*, 2016) proposed the measurement of concentric symmetry, that is defined in two different ways: backbone symmetry (Sb_i) and merged symmetry (Sm_i) . In Figure 8, we illustrate an example of the symmetry measurement calculations for the red node. The backbone pattern is created by removing the edges between nodes in the same hierarchical level. Next, the merged pattern is constructed by merging connected nodes into a single node. Additionally, we show, for each node, the probability of reaching it in a random walk through the graph.

Figure 8 – Examples of symmetry calculations for the red node. In (a), there is the original network. In (b), the symmetry backbone after the network transformations. Finally, in (c), the symmetry merged after the network transformations.



Source: Elaborated by the author.

Finally, the concentric symmetry (TRAVENÇOLO; COSTA, 2008) $S_i^{(h)}$, of a node i for a concentric level h, can be computed by:

$$S_i^{(h)} = \frac{\exp\left(-\sum_{j\in\Gamma_h(i)}(p_{ij}^{(h)} \times \log p_{ij}^{(h)})\right)}{|\Gamma_h(i)| + \sum_{r=0}^{h-1}\eta_r},$$

with η_r being the number of nodes that are not connected to the next concentric level $(h+1)$, $\Gamma_h(i)$ being the set of neighbors of i that also belong to level h and $p_{ij}^{(h)}$ as defined above.

2.4.3 Clustering coefficient

In contrast, the clustering coefficient measurement is actually derived from observations of a specific property of real life systems: if A is related to B, and B is related to C, then there is a higher probability that A is also related to C. With that in mind, the clustering coefficient cc_i of a node is calculated as the ratio between the number of triangles (N_Δ) and the number of triples (N_3) connected to i:

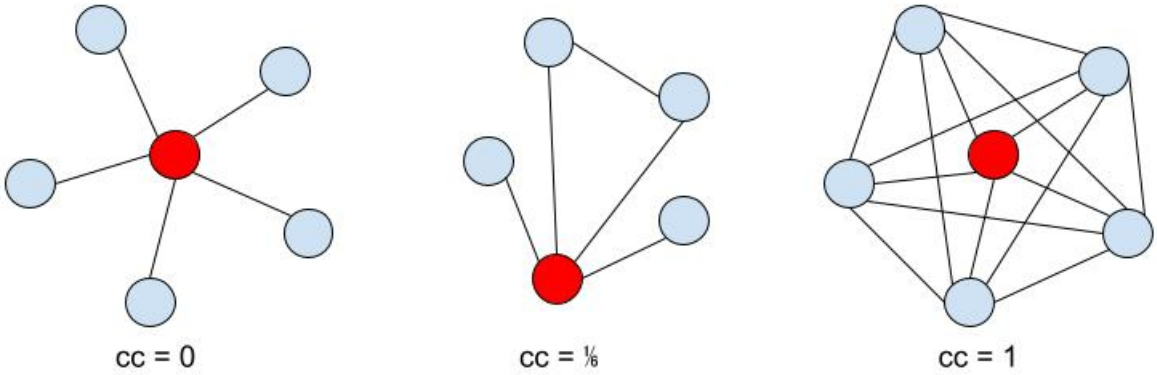
$$cc_i = \frac{N_\Delta(i)}{N_3(i)}$$

For that, a triangle is defined as a cycle of size three and a triple connected to i is any triple (i, j, l) where both j and l are connected to i. See Figure 9 for three examples of computations of a node's clustering coefficient value.

2.5 Modelling texts as networks

As discussed, complex networks are widely used to represent and study many different real-life systems, and one of these possible applications is the modelling of texts. When using complex networks to model this sort of systems, there are still many different possibilities. In

Figure 9 – Examples of clustering coefficient calculations for the red nodes. In the left, $cc = 0$, in the middle, $cc = \frac{1}{6}$ and, in the right, $cc = 1$.



Source: Elaborated by the author.

a small scale, one could represent every word of a text as a node in the graph, and the edges would be existing relationships between those terms, when considering the proximity in the text, for example, we get the definition of the *word adjacency network*, also known as co-occurrence networks. In contrast, on a larger scale, one could have a network in which each node is a full text, an article for example. In this scenario, there could be edges between articles that reference each other to construct a citation network.

In this Master's work, however, the focus is on an intermediate scale, following a less known mesoscopic approach by considering recurrence networks. In it, each node represents a window of paragraphs of a book (which optimal size was defined empirically, to assure that each element would have enough semantic context in it). To create the long range links, we create edges between similar nodes, as will be defined later on in this document. With this mesoscopic approach, one is able to capture important contextual properties from the text, what would not have been possible if working in any other scale. By considering a sequence of paragraphs, each element has enough semantic context (chraracters, location, actions, etc) to recur along the text and, hence, establish long range connections along the network and capture the narrative context that is told by that book. Therefore, what is proposed in this project is an innovative technique to model the semantic meaning of a text by using a recurrence network modelling.

CHAPTER

3

NATURAL LANGUAGE PROCESSING

3.1 Introduction

The study of linguistics is a very broad field that studies natural languages, that is, the forms of human communication that emerged spontaneously. The components of a natural language are phonetics, phonology, morphology, lexicon, syntax, semantics and pragmatics (BRISCOE, 2014), and this research focuses in two of them: syntax and semantics. Syntax is about the combination of words in a certain order to form grammatical sentences, where each of them play a specific role in that construction (e.g. verb, subject, object, and others). Semantics, on the other hand, concerns the combination of the individual meaning of each term to form a larger, common semantic context of the sentence as a whole.

Within the linguistics scope, the study of Natural Language Processing consists of the automatic processing of these natural languages. This area is present in many forms, such as text classification (ARRUDA; COSTA; AMANCIO, 2016), authorship recognition (AMANCIO, 2015b); (MEHRI; DAROONEH; SHARIATI, 2012); (SEGARRA; EISEN; RIBEIRO, 2015), sentiment analysis (ESTRADA, 2011), stylometry (AMANCIO, 2015a) and others. It has continuously received more attention and investment, accompanying the increase in data availability and the emergence of new techniques to tackle different sorts of problems. Thus, in this chapter, we will present some of the basic well known NLP concepts and techniques that will be important for this Master's work.

3.2 Text pre-processing

Text pre-processing is usually the first step applied when solving an NLP tasks, and it consists of cleaning up the text. When tackling a specific problem, there are some elements of the language that are usually not useful or even harmful, and this is where the algorithm can act on and mitigate them. One of the most common techniques of text pre-processing is the

Table 1 – Text pre-processing pipeline example.

Original text	"We are studying languages."
Punctuation removal	"We are studying languages"
Case treatment	"we are studying languages"
Tokenization	["we", "are", "studying", "languages"]
Lemmatization	["we", "be", "study", "language"]
Stopwords removal	["study", "language"]

tokenization, which consists of splitting the text into a list of tokens. Usually, the text is split by white spaces and, hence, a token is generally just a word. This procedure is widely employed because it is usually easier to deal with a list of terms instead of working with the text as a whole. A term has a specific syntactic classification and a semantic meaning that can be consulted in existing dictionaries, which can be very useful when tackling many sorts of NLP problems.

Another well known technique is punctuation removal, which consists of ridding the text of non-alphanumeric characters, such as ".", "_", ";", and others. It is widely used because punctuation marks usually do not carry any syntactic or semantic meaning and, therefore, are negligible for the majority of the NLP problems. Likewise, there is also *stopwords* removal. *Stopwords* are words that carry little or none contextual meaning, such as prepositions and articles. These terms are very common in any type of text or context and, consequently, can be misleading to some approaches that depend on the semantics of the words.

Additionally, it is also important for text processing to consider the case treatment. It is very common to consider every token in its lowercase format, to make sure that a word will always be represented by a single token, independently of the casing that it appears in the text. With that in mind, there is also the *lemmatization* technique (MANNING; SCHÜTZE, 1999), which is used to normalize terms to their canonical forms by disregarding gender and numeric inflections, for example. Finally, with all these steps for text processing, we construct a pre-processing pipeline, which is commonly employed in the text at the beginning of any NLP task (see Table 1 for an example of that pipeline).

3.2.1 Co-reference resolution

Another more sophisticated technique, but also commonly used in NLP problems, is the co-reference resolution (MANNING *et al.*, 2014). Co-references are expressions that refer to a previous mentioned entity in the text, i.e. pronouns. When applying co-reference resolution, one rids the text of all these references, replacing it by the actual entity being referred to, which is usually a subject. Take for example this sentence from "Alice in Wonderland", where Alice refers to the Mock Turtle: "Alice thought to herself, “I don’t see how he can ever finish, if he doesn’t begin.”". After applying this technique to the text, it would yield: "Alice thought to **Alice**, “**Alice** don’t see how **Mock Turtle** can ever finish, if **Mock Turtle** doesn’t begin.”".

3.3 Bag of words

When solving NLP problems that deal with multiple documents (e.g. text classification), it is very common to employ the Bag-Of-Words (BOW) model (BUNT; BOS; PULMAN, 2013) to represent the texts. It consists of transforming texts into word occurrences vectors, that is, every document is modelled as a vector, where each position maps to a specific word of the vocabulary, and the value in place is the number of times that term occurred in the document. Thus, each of these vectors' sizes is the total number of distinct words in D (that is, the set of all documents), and they work as a feature array for one specific document.

However, it is not mandatory for each position of the vector to refer to one specific word. Another methodology, for example, considers a set of n adjacent words, called *n-grams* (FELDMAN; SANGER, 2006), and, by that, the approach can also take into consideration the order that the words appear in the text. See Figure 10 for an example of a transformation of a set of documents into their BOW representations.

Figure 10 – Example of a set of documents and their respective Bag-Of-Words representations, considering 1-grams and 2-grams.

Documents

D1: Research is interesting
D2: I like to research
D3: It is interesting to read

Standard Bag-Of-Words (1-grams)

	D1	D2	D3
I	0	1	0
interesting	1	0	1
is	1	0	1
it	0	0	1
like	0	1	0
read	0	0	1
research	1	1	0
to	0	1	1

2-grams Bag-Of-Words

	D1	D2	D3
I_like	0	1	0
interesting_to	0	0	1
is_interesting	1	0	1
it_is	0	0	1
like_to	0	1	0
research_is	1	0	0
to_read	0	0	1
to_research	0	1	0

Source: Elaborated by the author.

3.3.1 TF-IDF

Another well known derivation of the BOW method, is to employ Term Frequency-Inverse Document Frequency (TF-IDF) to the document vectors. With this approach, instead of simply considering the frequency, each position of the vector quantifies the importance of each word. This importance of a word w , in a given document d , for a set of documents D, is computed by $\text{tf-idf}(w,d,D) = tf(w,d)idf(w,D)$, where tf is the *term frequency* and is computed by

$$tf(w,d) = \frac{f_w^d}{f_w^D}, \tag{3.1}$$

and idf is the *inverse document frequency* and is computed by

$$idf(w,D) = \log \frac{|D|}{N_w}, \tag{3.2}$$

considering that f_w^d is the frequency of word w in document d, f_w^D is the total frequency of word w in all documents in D and N_w is the number of documents in D in which the word w occurs.

3.3.2 Cosine similarity

After transforming the documents into vectors, one must have a way to compare them, and a very common method for doing that is to use the Cosine similarity (*sim*) . The reason behind it is that, when considering this BOW approaches, it is very likely that most of the vectors are sparse, that is, have a lot of zeros in them. Therefore, to avoid considering two vectors with many zeros similar, it is possible to employ the cosine measurement, that will ignore all the zeros while computing the similarity value.

It is known that the cosine of an angle θ between two vectors A and B can be computed as:

$$\cos\theta = \frac{A \cdot B}{\|A\|\|B\|}. \tag{3.3}$$

Therefore, the cosine similarity *sim* can be written as:

$$sim(A,B) = \frac{\sum_{i=1}^{n} A_i B_i}{\sum_{i=1}^{n} A_i^2 \sum_{i=1}^{n} B_i^2}. \tag{3.4}$$

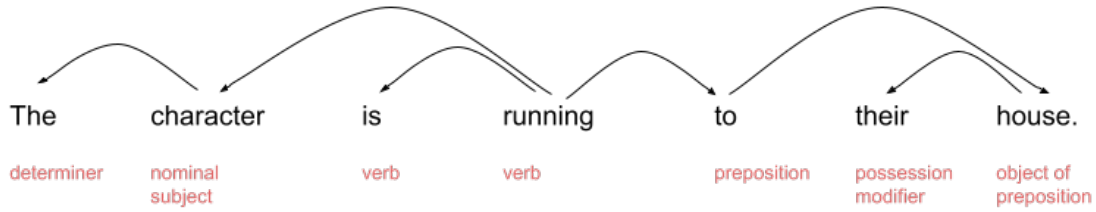
It is important to notice that this value is already normalized between 0 and 1, since the vectors only contain positive values and, therefore, $\theta \in [0, \frac{\pi}{2}]$.

3.4 Syntactical dependency analysis

As mentioned earlier, the syntax of a language concerns the relations between the words that form a sentence. Therefore, any sentence can be parsed into a dependency tree that names the syntactic function for each of its words and presents the dependency relations between them. The phrase *"The character is running to their house"*, for example, can be parsed into the dependency tree shown in Figure 11. This dependency parser is implemented and available in the spaCy Python library (SPACY..., 2016).

This technique is so important because it is also able to grasp a little of the semantics of a sentence. For the example in Figure 11, we know that the sentence has the verb "running" as its root, which tells the reader *what* is happening. There are also two important modifiers for this verb, "character" and "house", that represent, respectively, *who* is performing that action and *where* that happens. Therefore, by focusing on specific syntactic classes, in this case *verb*,

Figure 11 – Example of the syntax dependency tree for the sentence *"The character is running to their house"*.



Source: Elaborated by the author.

subject and *object*, it is possible to capture the contextual meaning of a sentence, which will be very important for our work.

3.5 Doc2vec

Recently, new research has been developed based on Neural Networks to represent and classify texts and documents. Two important examples of this work are the word2vec technique, defined in (MIKOLOV *et al.*, 2013b), and the doc2vec (LE; MIKOLOV, 2014), which was thought on top of the former definition. These approaches aim at representing texts through dense vectors that can capture short-range relationships between words, which can further be used to classify sentences or even whole documents.

One of the most interesting aspects of these techniques, is that the vectors yielded clearly represent semantical relationships between words, also facilitating the comparison between them. For example, given their word2vec representation, *Paris* - *France* + *Italy* = *Rome*. This outcome is really interesting when assessing NLP problems, specially when it is desired to capture the semantic context from texts. Henceforth, this technique will also be discussed later in this Masters' research.

CHAPTER

4

RELATED WORKS

In the literature, there are numerous works that address the text processing problem. This chapter will present four of those that have been studied, due to their important contributions for this research. Amongst the main topics discussed in these articles, we highlight: i) the modelling of texts as networks; ii) the application of syntactic knowledge to enhance informativeness; iii) the story flow analysis based on text entities; and iv) the employment of a mesoscopic scale to grasp a higher level context of a narrative.

4.1 Work of (ARRUDA; COSTA; AMANCIO, 2016)

In (ARRUDA; COSTA; AMANCIO, 2016), it is proposed a methodology focused on the structural properties of the text instead of its semantics. The network modelling is done by considering the *stopwords* (also referred to as *function words*), and creating a co-occurrence network from the text. They discuss their results with a supervised classification to discriminate between informative and imaginative texts. Several different measurements are extracted from the constructed networks (symmetry and accessibility proving to be specially relevant) and the classification task yields very satisfactory results, overcoming other similar networked approaches.

Likewise many other NLP problem approaches, the authors begin by pre-processing the text. However, they foresee both options of either keeping or removing the *stopwords* from the text at this step, which is further discussed later on in the article. Subsequently, the network is constructed by following the usual word adjacency approach and, to characterize it, the authors propose a set of measurements capable of capturing particular textual properties of the text from the topological structure of the network.

Afterwards, the results are evaluated considering three different strategies: global without *stopwords* (GS), local without *stopwords* (LS) and local with *stopwords* (LSS). For the global

approach, the local measurements are summarized to represent the network as a whole, while for the local strategy, the set of individual measurements is considered. The authors also considered three classifiers: K-Nearest Neighbors (KNN), Classification and Regression Tree (CART) and Naive Bayes.

Finally, the best performance obtained was with the LSS, which yielded the higher accuracy rate for every classifier considered. Thus, the authors conclude that local measurements are actually more useful to the informativeness of the topological strategy. Specifically, the accessibility and symmetry measurements proved to be specially significant for a better performance of the classification task considered.

4.2 Work of (AMANCIO *et al.*, 2012)

In (AMANCIO *et al.*, 2012), the authors employ statistic-based methods to extract complex networks metrics to tackle the problem of automatic summarization for Brazilian Portuguese. Their method yields a very high Rouge score (LIN; HOVY, 2003) when evaluating the summarizer and an equally satisfactory precision to extract keywords from the text. They also consider the application of syntactical knowledge to their approach and its influence on the results obtained.

The first step of the methodology is creating a *word adjacency* network based on the pre-processed text. They created a weighted graph, where each edge weight reflects the number of times the connected words appeared adjacent to each other in the text. Additionally, syntactical knowledge is incorporated to the methodology to add more edges to the network, considering the syntactical dependency relations between terms.

Next, statistical metrics are extracted from the network in order to characterize its topology, namely: diversity centrality, vulnerability, betweenness, strength centrality, global efficiency, and others. These measurements, combined with random walk techniques, are used to measure the informativeness of each node in the graph and, therefore, capture the most important words of the text.

Finally, to evaluate their performance, the authors use the Rouge score, precision and recall. The proposed methodology without considering the linguistic knowledge already yielded great results for the considered corpus, for both automatic summarization and keyword extraction. When taken into account, the syntactical dependencies provide only a small enhancement of the results for both problems, indicating that, even though there is room for improvement, it might only be possible by considering more sophisticated techniques, such as semantic information.

4.3 Work of (LIU *et al.*, 2013)

In (LIU *et al.*, 2013), it is proposed an optimized method (*StoryFlow*) to generate an aesthetically appealing story line visualization that considers the hierarchical relationships between the text entities over time. By ordering and aligning the entities and representing the interactions between them, their approach enables users to better understand the contextual meaning of the narrative and how it evolves.

StoryFlow's layout pipeline consists of four steps: i) hierarchy generation; ii) ordering; iii) alignment; and iv) compaction. The first stage is responsible for constructing the dynamic relationship trees for the various events that take place throughout the different periods of the story. Next, entity lines ordering and alignment is employed to minimize line crossing and facilitate the reading and understanding of the visualization. Lastly, and with the same intent, an optimization technique is applied to minimize the white spaces and wiggle distance in the layout.

Subsequently, the authors further explain the optimizations and techniques used to compact the produced layout and make it more aesthetically appealing. However, this discussion is not considered in this Master's work, since the focus here is only how the text entities and the relationships between them are used to represent the text's narrative.

Finally, to validate their method, several different experiments are considered, where the authors evaluate the similarity to hand-drawn designs and some metrics such as line crossing, white spaces and wiggle distance to measure success. Furthermore, they also gather user feedback that corroborates the quality of the layout generated by *StoryFlow*.

4.4 Work of (ARRUDA *et al.*, 2016)

In (ARRUDA *et al.*, 2016), the authors address some limitations encountered in word adjacency networks, including the absence of a community structure and the consequent inability to portray the topical structure of a text. To tackle this issue, they propose a mesoscopic approach to represent texts as networks, in which a node represents a larger context of the text, namely, a sequence of subsequent paragraphs. As a result, the methodology is able to reflect the story flow of a narrative, taking into consideration its semantic context.

The first step for the methodology described is to pre-process the text, which includes punctuation, numbers and *stopwords* removal, followed by applying a lemmatization technique to the text. Thus, after splitting the text into a sequence of paragraphs, the organized text $O = [p_1, p_2, ...p_n]$ is obtained. From that, the paragraph window for each network node is extracted considering Δ subsequent paragraphs.

Next, to create the edges, the authors employ the Bag-Of-Words technique, combined with TF-IDF to represent every node as a vector and, then, compute the similarity between every pair of nodes using the cosine similarity. Thus, a fully connected network is obtained, where the

edge weights are the similarities between the two connected vertices. Then, only the edges with weight higher than a threshold T are maintained and, afterwards, the weights are disregarded for the remaining of the work.

To evaluate the results, the authors propose a case study of the book "Alice's Adventures in Wonderland", where they analyze the story flow of the narrative and how that is portrayed in the constructed network. For that, they considered a force-directed network visualization technique, based on the Fruchterman-Reingold (FR) (FRUCHTERMAN; REINGOLD, 1991) algorithm, that naturally highlights some topological properties of the network displayed. With that analysis, they confirm that the network constructed displays a chain-like structure, that portrays the story flow of the book. Additionally, they observe that connections between distant nodes indicate regions of the book with a high contextual similarity.

Finally, the authors discuss the results obtained when differentiating real texts from meaningless ones (shuffled texts). For that, they extract several different measurements from the networks and compare their values for both text categories, finding that they are significantly different. The authors also compare their proposed technique with standard co-occurrence networks by demonstrating their higher performance in the task of differentiating real and meaningless texts. Therefore, corroborating the hypotheses that the mesoscopic network is able to better represent the semantic context of a book's narrative.

CHAPTER

5

METHODOLOGY

This chapter describes the whole methodology pipeline proposed in this masters' research, which can be summarized in three main steps: text processing, network modelling and network characterization, that are briefly described as follows:

1. *Text processing*: the main goal in this step is to facilitate the analysis of the relationship between specific words with different syntactic roles, namely subject, verb and object. Therefore, a syntactic parsing algorithm is applied with the purpose of identifying such relevant words. Additionally, stopwords and punctuation marks are also disregarded, since they do not add any semantic value to the text's narrative.

2. *Network modelling*: this step is responsible for modelling a text into its respective recurrence network. This constructed network is a semantic representation of the narrative flow, where its nodes represent a sequence of δ paragraphs and its edges are established according to the semantic similarity between the nodes.

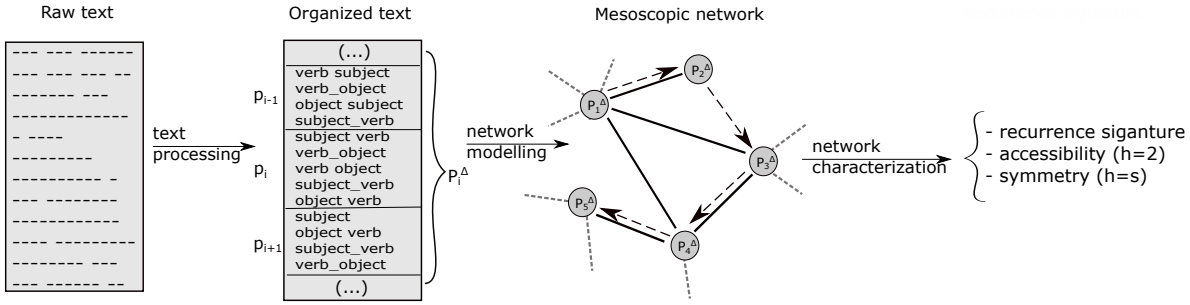
3. *Network characterization*: finally, this stage aims at characterizing the generated networks via different topological measures. Additionally, the Recurrence Signature is proposed as a new measure based on the recurrence of semantic context along the text.

Those steps are also summarized in Figure 12, a visual representation of this Masters' research pipeline.

5.1 Dataset

The dataset used in this project consisted of 300 different books. They were all retrieved from a random selection of the Project Gutenberg (PROJECT..., 1971), filtering only to those written in English that had between 1000 and 2000 paragraphs. Therefore, this selection was unbiased on any other aspect of the books, e.g. author, publish date and genre.

Figure 12 – Diagram of the execution pipeline of the methodology proposed in this Chapter. The pipeline comprises three main steps: text processing, network modelling and network characterization.



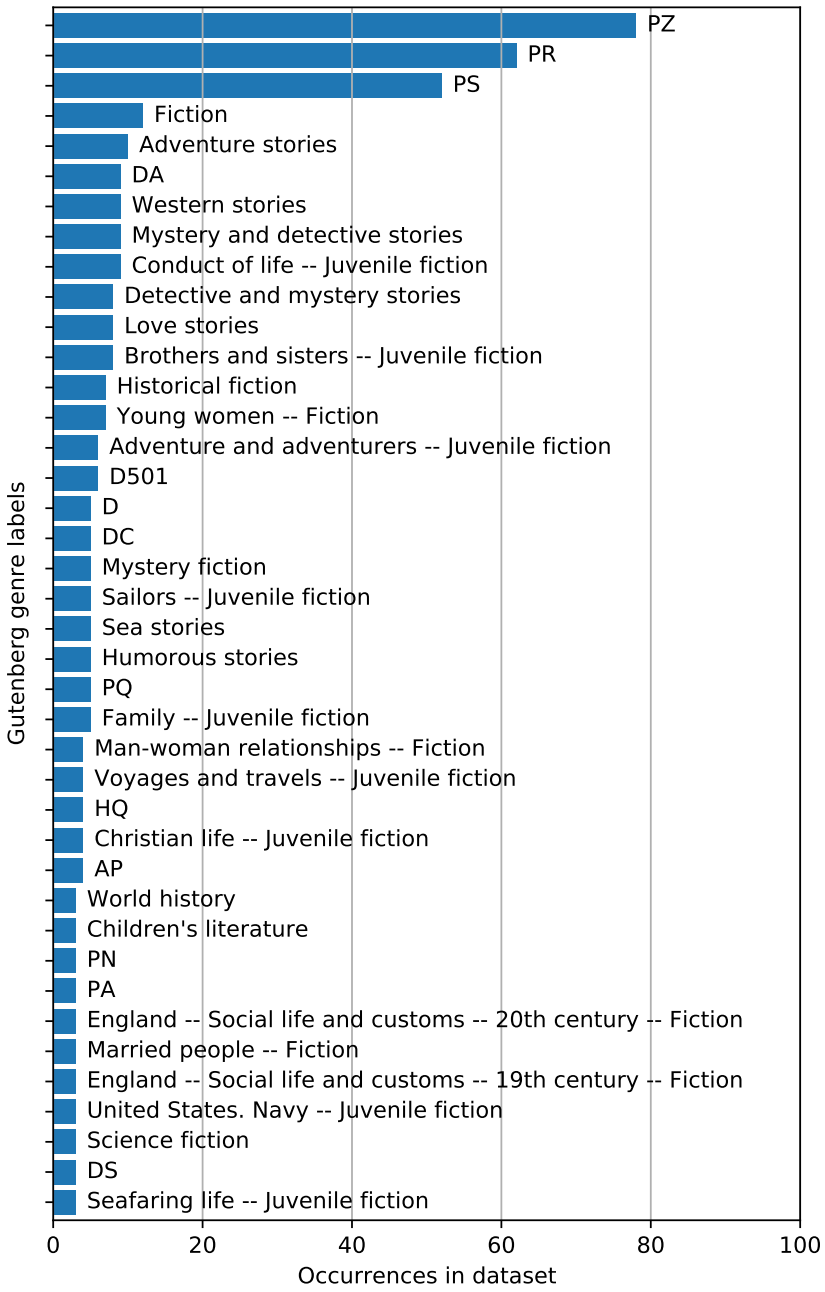
Source: Elaborated by the author.

The Gutenberg Project provides open access to more than 60 thousand books, written in several different languages. For each book, its content and some metadata is available, including: *author*, *illustrator*, *title*, *language* and *subjects*. The latter consists basically of a list of literary genres that can be assigned to its respective book, and that is why it will also be referred to as *genres* in this document. However, for this this research, the chosen dataset comprises only the raw text content and the set of genres for a given book.

There are important aspects of what Gutenberg provides as a book's set of subjects that should be disclaimed. Since there are multiple categories and they lack of any distinction of importance, there is no straightforward form of delegating one specific genre for each book and, therefore, there is no clear and well determined classification for a book's literary genre. Moreover, there is an extensive scale of different granularities for these categories. Some are very broad and imprecise, such as *PR (Language and Literature: English literature)* or *PZ (Language and Literature: Juvenile belles lettres)*. While others are very specific and restricted, such as *Scarecrow Fictitious character from Baum* or *National Research and Education Network Computer network*.

To illustrate this scenario, Figure 13 plots the occurrence distribution for the 40 most common genres in the 300 books dataset studied in this work. There are some classic literary genres, such as children's literature, love stories, fiction and its derivations. Nonetheless, the top two entries combined, 'PZ' and 'PR', are present in almost half of the dataset, even though they do not have a straightforward semantic interpretation such as the aforementioned categories. As one can notice, the classification problem for this dataset is a deeply complex problem and, henceforth, a parallel study of Gutenberg genres was proposed in this research, to address the dataset limitations in order to make classification experiments feasible, for example. This investigation is described later, in Section 6.3.1.

Figure 13 – Plot of the literary genres distribution found for the books in the Gutenberg dataset.



Source: Elaborated by the author.

5.2 Text processing

On any project that involves NLP, data collection, markup and annotation are important steps when working for solutions (INDURKHYA; DAMERAU, 2010). The researchers have to take into consideration the specificities of the problem they are trying to solve and evaluate how text processing can enable and facilitate possible solutions. In this Masters' research, particular decisions were made in order to deal with the proposed dataset, while optimizing the specifics goals of characterizing a text's semantic context by using recurrence networks.

As a first step, some pre-processing strategies are applied to reduce noise in the following stages. Since the books are in a raw text format, there are markers along the text (such as chapter markers, underscores ("_"), etc) that do not contribute with to the syntactical or semantic aspects of the text and, hence, should be removed. However, it is important to notice that we do not remove punctuation nor *stopwords* at this point, since those are important during the syntactical dependency analysis stage.

Additionally, a co-reference resolution technique (MANNING *et al.*, 2014) is applied in order to obtain better results when applying the syntactic analysis algorithm. In this step, one hopes to guarantee that only one term will be used to refer to each and every entity of the text, which is then beneficial to establish the relationships between them.

Finally, after the pre-processing step, the syntactic analysis is employed. At this point, each paragraph is reduced to a set of tokens with specific syntactic roles: either *subject*, *verb* or *direct object*. This role selection was chosen with the intention of getting only the tokens that provide the most contextual meaning to the sentences where they are found, while excluding any possible noise. This strategy was based on the fact that, to understand the semantics of something, it is usually necessary to answer the questions *what is happening* and *who is doing it*. To answer the latter, one can retrieve the term in a sentence that has the syntactic role of subject, since it is the actor responsible for action in that sentence. Secondly, when recovering the verb and its direct object, the action taking place is depicted, which answers the first inquiry. This approach is also supported by the fact that the incorporation of linguistic knowledge can contribute to text summarization (LEITE *et al.*, 2007), and, analogously, to capture the contextual meaning of texts by enhancing informativeness.

Ultimately, after the syntactic analysis step, the text is reduced to a set of tokens, all of which were either subjects, verbs, or direct objects of their respective sentences. Each of the generated tokens is then normalized to its canonical form by using a lemmatization technique (MANNING; SCHÜTZE, 1999), resulting in the disregard of inflections in verbal tense, number, case or gender. The lemmatization step ensures that two terms with the same meaning (such as a verb in a different time tense or a subject in plural tense) will only have one representation in the text, since they should represent the same entity.

To illustrate with an example, given the sentence "_thought Alice to herself_", it will first be transformed to "thought Alice to herself" (after marker removal), next to "thought Alice to Alice" (after co-reference resolution), then to "thought Alice" (after syntactic analysis) and, finally, to "think Alice" (after lemmatization), which is described in table 2. At this point, the semantic meaning of the sentence is summarized in the final set of tokens, where it is possible to answer *what* is happening (by the verb *think*) and *who* is taking that action (by the subject *Alice*).

Succeeding all this processing, one gets the final organized text O , as it will be called from here on. O is a sequence of paragraphs p : $O = (p_0, p_1, p_2..., p_n)$, where n is the number of paragraphs in the text. Each paragraph comprises a sequence of words w :

Table 2 – Text pre-processing pipeline applied for the example sentence "Thought Alice to herself".

Original text	"_thought Alice to herself_"
Marker removal	"tought alice to herself"
Syntactic analysis	["thought", "Alice"]
Lemmatization	["think", "Alice"]

$p_i = (w_{i0}, w_{i1}, w_{i2}..., w_{in_i})$, where n_i is the number of words in paragraph i. Ultimately, the recurrence network is built upon the organized text O.

5.3 Network modelling

In recent years, a new set of techniques has been introduced to create networks from documents, which takes into account their mesoscopic structure. In (ARRUDA *et al.*, 2016), for example, the methodology proposed addresses two important aspects typically overlooked by more traditional approaches: (a) the mesoscopic structure of a text and (b) its unfolding along time. This Masters' work proposes a new method to construct mesoscopic networks (which will be referred to as recurrence networks in this document) as a complementary extent of their research. The methodology presented here will still address the two aspects aforementioned and add a new consideration: the syntactical dependencies established along the text. This procedure to generate such networks will be explained in this section.

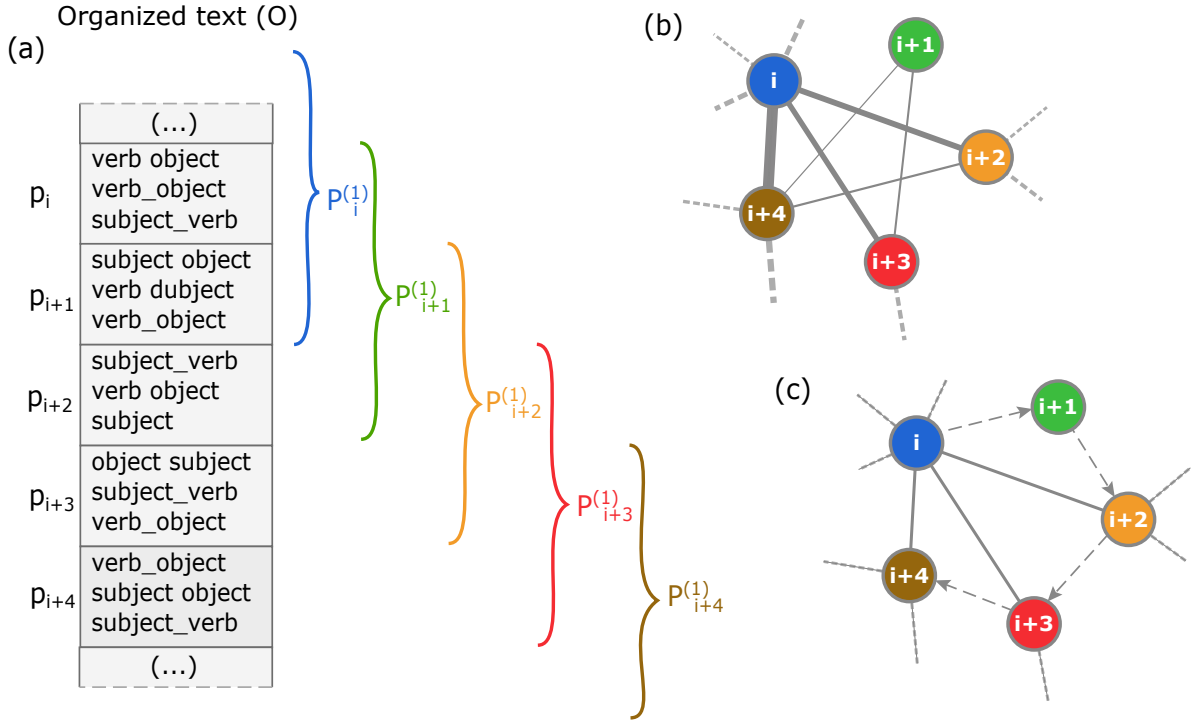
From the organized text O, one can get a sequence of paragraphs $P_i^{(\Delta)}$, such that $P_i^{(\Delta)} = (p_{i-\Delta}, p_{i-\Delta+1}, ...p_i, p_{i+1}, ...p_{i+\Delta})$, which will be referred to as a "paragraph window". Each paragraph window $P_i^{(\Delta)}$ has size $2\Delta+1$, and maps to one unique node of the final recurrence network. This procedure is illustrated in Figure 14(a), considering $\Delta = 1$.

To explain the methodology, let us first take paragraph p_{i+1} as a trivial example. For $\Delta = 1$, its corresponding paragraph window would be $P_{i+1}^1 = (p_i, p_{i+1}, p_{i+2})$, which would be mapped into the node in green, in Figure 14(b). This scenario is analogous to all the other nodes in the network.

Next, to create the edges, a technique combining tf-idf with *bag-of-words* is employed, as it was explained in Chapter 3. For this application, the set of documents D is equivalent to the collection of paragraph windows $P_0, P_1, P_2..., P_n$. Each document d_i is the text within one paragraph window P_i, and w is each one of the terms in the text, after applying all the pre processing steps. Finally, cosine similarity is used to calculate $sim(P_A, P_B)$ between paragraphs A and B, where $A, B \in V$ and $|A - B| > \Delta$. This latter restriction due to the fact that, otherwise, the two paragraph windows would share at least one paragraph and, therefore, the similarity computed between them would be biased.

As a result, a fully connected network is created (see Figure 14(b)), in which the edge weights correspond to the similarity $S(P_A, P_B)$ normalized between 0 and 1 among each pair of

Figure 14 – Illustration of the presented methodology on how to construct the recurrence network from on the organized text O. Initially, the paragraph windows P_i are extracted from O, each representing one node. This is illustrated in (a). Then, a fully connected weighted network is constructed, where the edge weights are the cosine similarity calculated between both nodes, which is illustrated in (b) by the line widths. Next, only the $|V| \times T$ strongest connections are maintained, so that the average degree of the graph is now equal to T. Finally, the sequence edges are added between consecutive vertices following index order, as illustrated by the dashed edges in (c).



Source: Elaborated by the author.

nodes. In the picture, the edge weight is represented by the line width. The final mesoscopic network is obtained by pruning the weakest connections until the average degree of the network reaches a specified threshold T. After this procedure, edge weights are ignored, resulting in an unweighted network (see Figure 14(c)) with a fixed average degree of T, given that $T \geq 0$. All of these edges will be referred to as *similarity edges* hereafter.

Finally, $|O| - 1$ edges are inserted in the recurrence network, in a way that they link nodes corresponding to adjacent paragraph windows, that is, node P_1 will be connected to P_2, P_2 to P_3 and so on, as illustrated in Figure 14(c) by the dashed edges. These edges are marked as *sequence edges*, and they are not considered when calculating network measurements in the future, but they will guarantee that the network is a connected component and will facilitate the employment of the network visualization technique, since they guarantee a sense of chronological order to the overall structure.

5.4 Network characterization

After all the text pre processing steps and the network modelling procedure, a recurrence network is yielded from the studied document. To analyze, extract knowledge from it and experiment with it, several different measures are proposed. They will be explained in detail in this section.

5.4.1 Network topology measures

There are several different measures with a great variety of motivations behind them in the literature. To successfully characterize a network topology, one has to determine which aspects are the most relevant to the problem in question and take them into account when choosing which shall and shall not be used. In this Masters' research, the main goal is to use recurrence networks to characterize a textual document from a mesoscopic perspective, while extracting from it its essential semantic context. For this task, two primary measures were chosen: accessbility and symmetry.

As it was explained in Section 2.4.2, the accessibility of a node proposes a quantifiable number to, given a limited, arbitrary distance, how many other nodes are effectively accessible from that specific starting point. Accordingly, one can say that this measure is appropriate to discriminate between peripheral or central nodes in a network, which can be useful for example, in NLP problems, identifying relevant words in texts and, hence, extracting keywords or identifying author stylometry (AMANCIO; SILVA; da F. Costa, 2015). Moreover, since the symmetry measure derives from the concept of accessibility, concentric levels and random walks it is also fit to grasp the topology heterogeneity in a recurrence network.

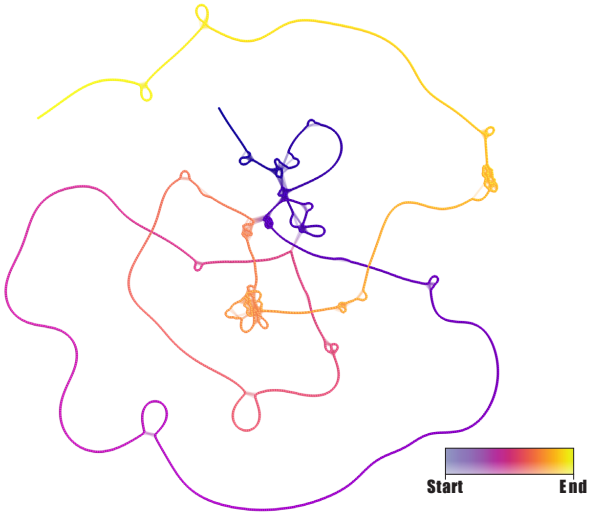
Another aspect considered when choosing this measures to characterize the recurrence networks proposed in this work is the fact that they are multi-scale. This means that one can adjust the path length in the calculations to consider a broader or narrower region of the network, based on each problem specificities. This is important when dealing with textual documents since they can have very different sizes, so it is more adequate to adapt the measure considered to one's specific problem. In this research, several different path lengths were empirically tested before choosing the one that would yield the best results while still being intuitively suitable to characterize the semantic textual patterns, as it is proposed here. These experiments and its results will be addressed later in this document, in Chapter 6.

5.4.2 Extracting a network's Recurrence Signature

On a bolder perspective, additionally to using this well established measures, a new one is also proposed here: the Recurrence Signature (RS) . It is intended to capture the overall structure of the network, inspired by the visualization technique using the force-directed nodes placement described in (ARRUDA *et al.*, 2016). With this approach, each network forms a continuous line,

which can be interpreted as the story told along a book. This line can be bent to approximate two different points of the book, indicating that the content of those parts are similar to each other. Therefore, the intention with this new proposed measure is to capture the connections that bend this line, at what point and how often they occur in the network. An example of this visualization can be found in Figure 15, where the recurrence network for the book “The Arabian Nights Entertainments" is portrayed.

Figure 15 – Visualization of the recurrence network generated for the book "The Arabian Nights Entertainments".



Source: Elaborated by the author.

From Figure 15, one can tell how the story starts in a certain point of space (points in blue in the picture), unfolds along time with several bents, turns and loops, and then finally ends in another location (points in yellow). This structure is created due to the similarity edges that exist between different and possibly distant nodes in the recurrence network. Since this visualization technique is force-based, it means that the more similarity edges exist between two vertices, the closer they should be placed in space.

By design, these connections between distant parts of the recurrence network are established by the similarity edges whenever the two nodes are sufficiently similar to each other (given their cosine similarity). Accordingly, the Recurrence Signature proposed by this Masters’ research aims to represent the occurence patterns of this connections and cross references for a given recurrence signature. For that, the RS is a series of numbers that quantify the distance between every other cross reference present in the network, in other words, how many sequential nodes exist in the network until the next one that has at least one similarity edge linked to it. Algorithm 1 illustrates how to obtain the RS for a given network, by using the *EstablishesCrossReference* method described in Algorithm 2.

In Figure 16, on the left, it is portrayed an example network and the algorithm execution over it. Initially, $RS = $, the counter is set to 0, and the analysis start in vertex 1. Since it does

Algorithm 1 – Algorithm to construct RS_1

```
1:  procedure GETRS(g)                                   ▷ Getting RS for a graph g
2:      arr ← []
3:      cnt ← 0
4:      for v ∈ V do
5:          cnt ← cnt + 1
6:          if EstablishesCrossReference(v) then
7:              arr ← arr + cnt
8:              cnt ← 0
9:          end if
10:     end for
11:     return arr                                       ▷ RS is arr
12: end procedure
```

Algorithm 2 – Algorithm to find if that node establishes a cross reference

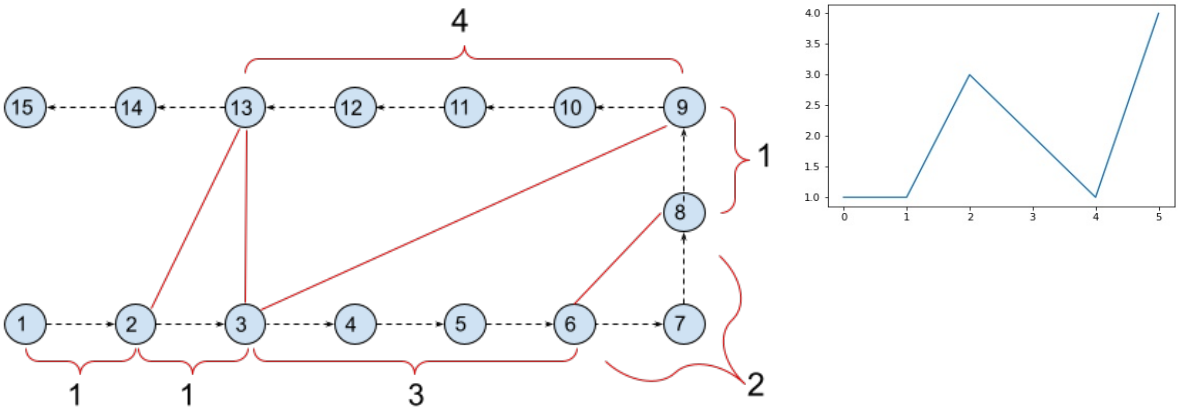
```
1:  procedure ESTABLISHESCROSSREFERENCE(g, v)   ▷ Returns true if there is a similarity
    edge conecting to vertex v of graph g and false otherwise
2:      for e ∈ E do
3:          if e.type = "similarity" AND (e.target = v OR e.source = v) then
4:              return true
5:          end if
6:      end for
7:      return false                          ▷ There is no similarity edge to/from vertex v
8:  end procedure
```

not have any similarity edges, it just moves on to the next iteration. After going to node 2, the counter is incremented by 1. Since there is a similarity edge there, that connects to a distant node and, hence, established a cross reference, the value of counter is appended to RS (now $RS = [1]$) and the counter is set back to 0. After getting to vertex 3, the counter is incremented again and, since there is another similarity edge connecting to that node, another 1 is added to the array ($RS = [1,1]$). Then the counter is set back to 0 and the iteration continues to the next vertex.

The algorithm stops when reaching node 15, the end of the graph, where the final signature $RS_1 = [1,1,3,2,1,4]$ is retrieved. Since the RS consists of a series of numbers, another way to visualize and interpret this measure would be through a chart. In this approach, the values of the series go into the y axis sorted by their indexes, which are on the x axis. The resulting chart for the example network is also shown in Figure 16, on the top right corner.

Each of these signatures is capable of representing the network itself and, essentially, the narrative constructed along a book. In Figure 17, one can see the signature plot of the *RS* for the book *The Arabian Nights Entertainments*, where it is possible to observe some specific properties of the narrative, such as the frequency of the cross references and how long it takes for another one to happen. Therefore, this series can be used to represent a specific network and, moreover, a specific book, as it works as a signature of the network itself. Hence, this measure

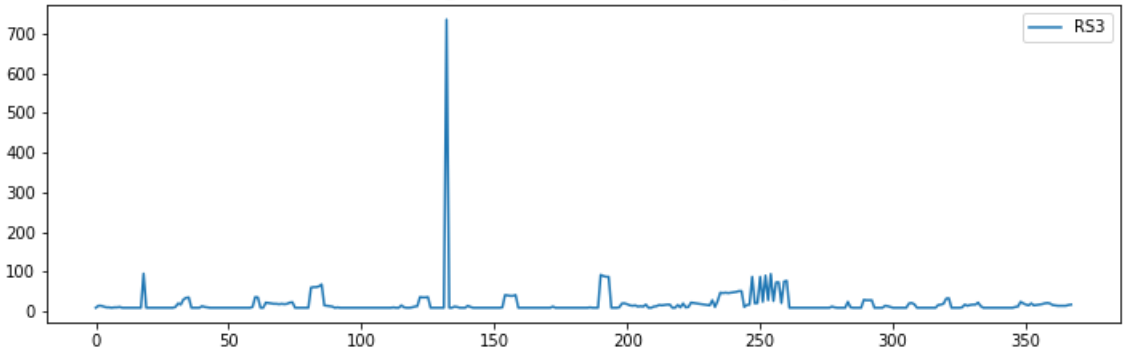
Figure 16 – Visual representation of the proposed recurrence signature *RS*. On the left, an example graph and the measure extraction from it. On the right, a plot of the generated Recurrence Signature, where the values are in axis y and the indexes in axis x.



Source: Elaborated by the author.

can potentially be used to solve different tasks, such as literary genre classification and real texts discrimination, which will be discussed in the following sections.

Figure 17 – Visual representation of the RS for the book *The Arabian Nights Entertainments*. Plotted using the *matplotlib* python library.



Source: Elaborated by the author.

CHAPTER

6

RESULTS

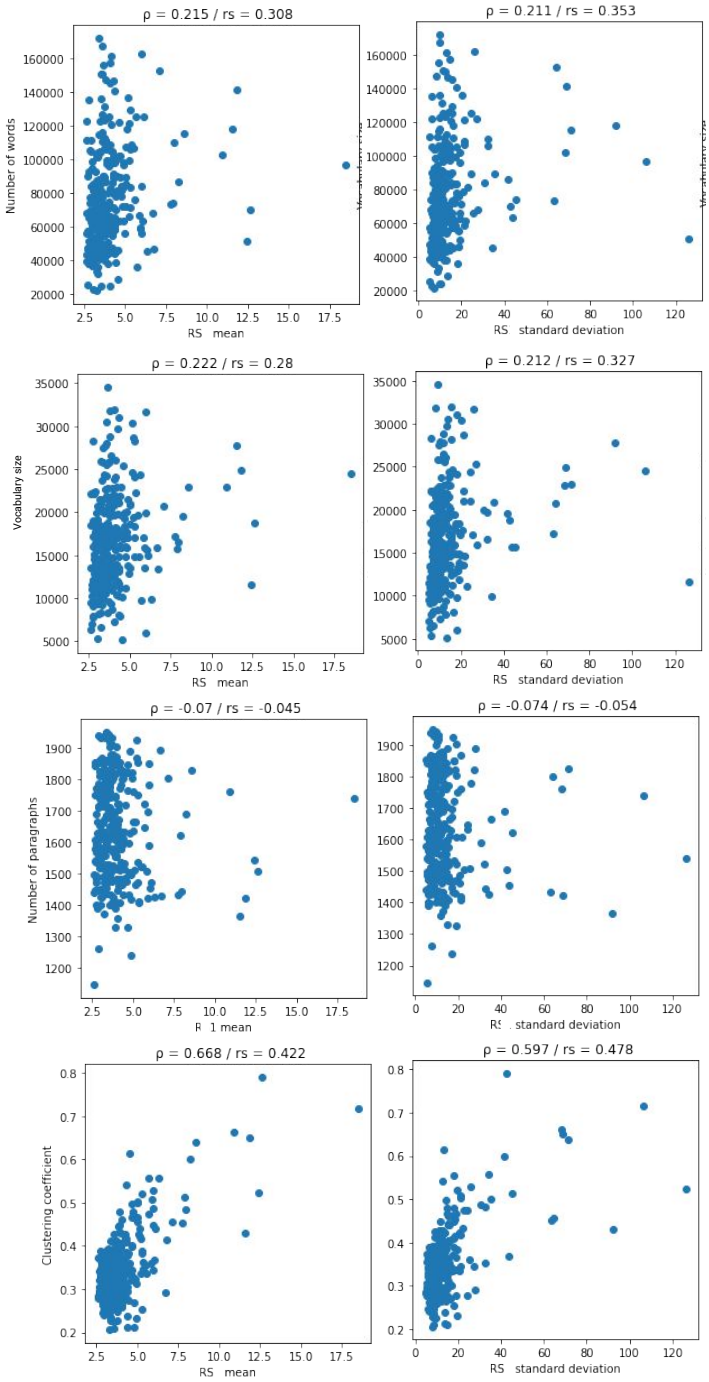
To evaluate the success of the methodology in meeting the established goals, four different experiment studies are proposed for analysis: i) Comparison with trivial network measures; ii) Discrimination between real and shuffled texts; iii) Literary genre discrimination; and iv) Comparison to other orthodox approaches. This chapter will be responsible for explaining the specificities of the experiments performed and the results yielded by each of them.

6.1 Comparison with trivial measures

One important aspect of the methodology of this Master's work is that the proposed measures cannot be redundant to other already existing ones. Hence, as an initial validation the Recurrence Signature measure presented, it was analyzed upon other trivial measures, namely: i) text size (total number of words words); ii) vocabulary size (number of distinct words); iii) number of paragraphs; and iv) network's clustering coefficient. Figure 18 shows the correlation plots for these measurements against the RS mean (on the left column) and the RS standard deviation (on the right column). Above each plot, there is the Pearson (ρ) and Spearman (r_s) correlation values for those pairs of measurements.

First, let us observe the first three, most trivial measures: text size, vocabulary size and number of paragraphs. The highest values found were $\rho = 0.222$ for the pair {RS mean, vocabulary size} and $r_s = 0.353$ for the pair {text size, RS standard deviation}. These values are not expressive enough to state any significant correlation between them, which is enough to say that they are, at least, not redundant. Finally, by analyzing the correlation of the pair {Clustering coefficient, RS mean}, one can observe that the pearson correlation value of $\rho = 0.668$ is indeed not negligible. However, the plot indicates that this correlation is not as clear as it could be. There is a kernel of data points centralized at the bottom left of the chart, and then a minority of them create the pattern of a linear relation, which probably yields the pearson value encountered. Additionally, since the clustering coefficient is a less trivial measure, a light correlation as

Figure 18 – Correlation plots for each of the trivial network measures considered. Each plot also informs the Pearson (ρ) and Spearman (r_s) correlations. Generated using the *matplotlib* Python library.



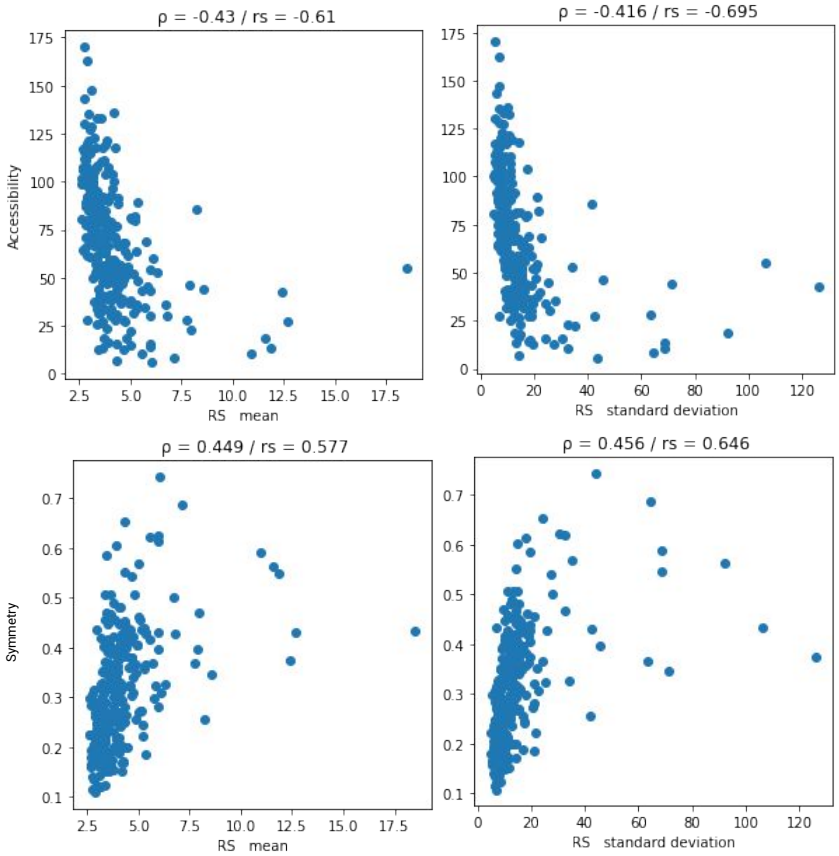
Source: Elaborated by the author.

established by this pattern is not too harmful for the research at this point. The Recurrence Signature could still grasp sufficiently different and enriching aspects of a text, as it will be discussed in detail hereon the following sections.

Similar to Figure 18, Figure 19 shows the same correlation plots, but now comparing the RS to accessibility and symmetry. One can observe that there are no significantly expressive Pearson values found, the highest one being $\rho = 0.456$ for the {Symetry, RS standard deviation}

pair. As for the Spearman correlation, the greatest value was found between {Accessibility, RS standard deviation}, with $r_s = -0.695$. Similar to the clustering coefficient case, having a light correlation between the proposed RS and the accessibility and symmetry measures is not an impairment to the research here. Accessibility and symmetry were recently proposed in (TRAVENÇOLO; COSTA, 2008) and (SILVA *et al.*, 2016) and are not trivial nor completely known in the research community. Hence, it is possible to state that the proposed recurrence signature is aligned with fairly recent proposals and potentially reflecting innovative topological properties of the network, which is our ultimate goal.

Figure 19 – Correlation plots for the measures chosen to characterize the networks studied in this work: RS mean and standard deviation versus accessibility and symmetry. Each plot also informs the Pearson (ρ) and Spearman (r_s) correlations. Generated using the *matplotlib* Python library.



Source: Elaborated by the author.

6.2 Discrimination between real and meaningless texts

To sustain that the proposed methodology's capability of capturing the contextual meaning of a narrative, the experiment from (ARRUDA *et al.*, 2016), to discriminate between real and meaningless texts, was reproduced. To create the dataset of meaningless texts, the paragraph order of the each of the texts in the original dataset was shuffled. From that, 300 new texts were added to dataset, each corresponding to one of the original books from Gutenberg. It is important

to notice that, since only the shuffled paragraphs order was shuffled, these new texts had the exact same content as its equivalent original version: same vocabulary, same size, even the same syntactical sense in every individual sentence of the text. However, with this kind of shuffling, the sense of a continuous narrative in the book was completely broken, since the reordering of the paragraphs rids the whole content of an overall semantic narrative, which, in the end, leaves only a meaningless text.

With this extended version of the dataset, the same methodology was applied to the new texts in order to generate the recurrence network respective to each one of them. Finally, two different experiments were performed with the same goal: use the recurrence network structure to differentiate between real and meaningless texts. The two approaches here will be to i) use accessibility and symmetry; and ii) use the Recurrence Signature. Both experiments will be described, respectively, in the following two sections.

6.2.1 Using topological measures: accessibility and symmetry

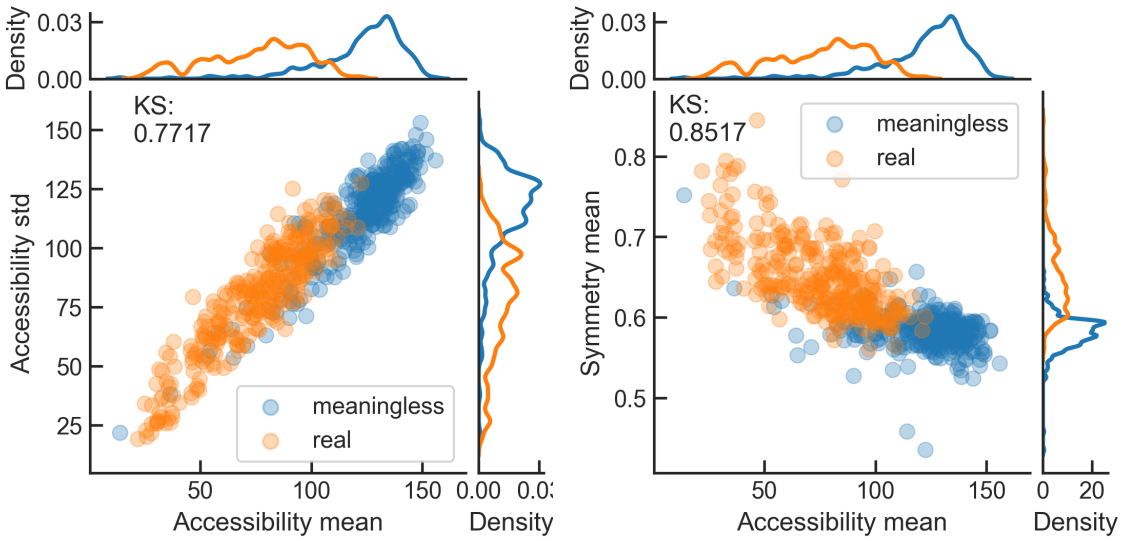
With the extended version of the dataset, three different measures were considered in order to characterize one specific network:

- mean of the accessibility (calculated for concentric level $h = 2$) found for every node: $mean_\alpha = mean(\{\alpha_1^{(2)}, \alpha_2^{(2)}, ..., \alpha_n^{(2)}\})$
- standard deviation of the accessibility (calculated for concentric level $h = 2$) found for every node: $std_\alpha = std(\{\alpha_1^{(2)}, \alpha_2^{(2)}, ..., \alpha_n^{(2)}\})$
- mean of the backbone concentric symmetry (calculated for concentric level $h = 2$) found for every node: $mean_S = mean(\{S_1^{(2)}, S_2^{(2)}, ..., S_n^{(2)}\})$

Given the proximity between the topological characteristics of the networks in question and the small-world model, choosing a too high value for h would mean that any random walk starting from an arbitrary point of the network could potentially sweep the entire network at once. This would be detrimental to the experiment, since it would make it more difficult to differentiate between each individual node of the network. Thus, it is valid to state that smaller values of h are more appropriate for this case scenario. By taking this into consideration and by performing empirical tests, it was established that a concentric level of $h = 2$ yielded the best results, and, therefore, it was the value chosen for the experiment.

With these three measures, two plots were generated, and they are both illustrated in Figure 20. For both charts, each datapoint represents one of the texts in the dataset, colored by their labels: orange for the original version, and blue for the randomized one. On the left, the data is plotted onto the accessibility mean values ($mean_\alpha$) on x axis and the accessibility standard

Figure 20 – On the left, discriminating between real and meaningless texts using accessibility's mean and standard deviation extracted from the recurrence network, with a high KS, equals to 0.7717. On the right, the same discrimination task, but now using accessibility and symmetry means, also with a high KS, equals to 0.8517.



Source: Elaborated by the author.

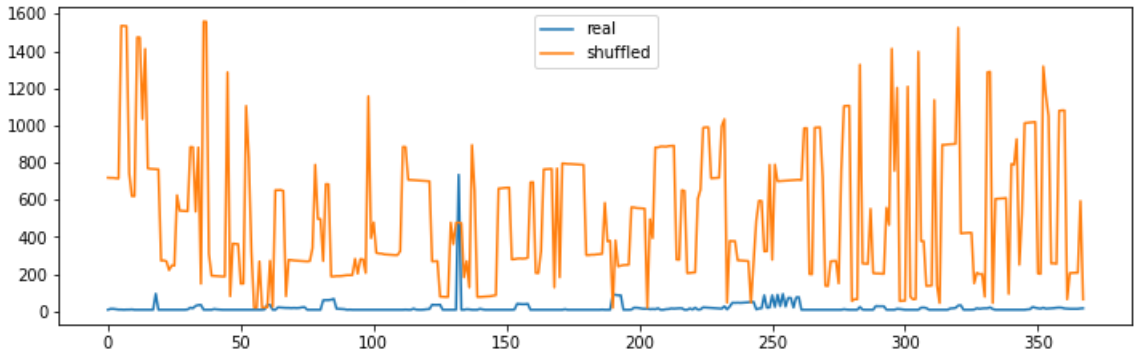
deviation (std_α) values on y axis. On the right, x axis still holds the accessibility mean ($mean_\alpha$), but now the data is plotted against the symmetry mean ($mean_S$) on y axis.

By pure observation of the actual plots and their density distributions, one can suppose that there is indeed a separation between the two groups in both plots. To further sustain this point, the statistical score of Kolmogorov-Smirnov (KS) (MASSEY, 1951) is used to compare the distributions. The Kolmogorov-Smirnov test is used to quantify the goodness of fit between two distributions. It can be used to tell how much they differ, the KS score being higher the most different they are. For both comparisons, a high KS score was yielded, namely $KS = 0.7717$ for the first and $KS = 0.8517$ for the second. All this contributes to say that the application of recurrence networks topology measures can successfully discriminate between real and meaningless texts, thus being able to capture a narrative's semantic context.

6.2.2 Using a network's Recurrence Signature

For the second experiment, the Recurrence Signature measurements were extracted from all the 600 recurrence networks (referring to both the original and shuffled version of every one of the 300 books). Figure 21 portraits an example plot of both extracted signatures for the book "The Arabian Nights Entertainments", the original (real) version in blue and the meaningless (shuffled) version in orange. It is visible how the structure of the two plots is essentially different, given that the randomized one is considerably more chaotic when compared to the pattern encountered in the one from the real text.

Figure 21 – Comparison between the recurrence signature for the book *The Arabian nights entertainments*' real text (in blue) and its shuffled version (in orange). Generated using the *matplotlib* Python library.



Source: Elaborated by the author.

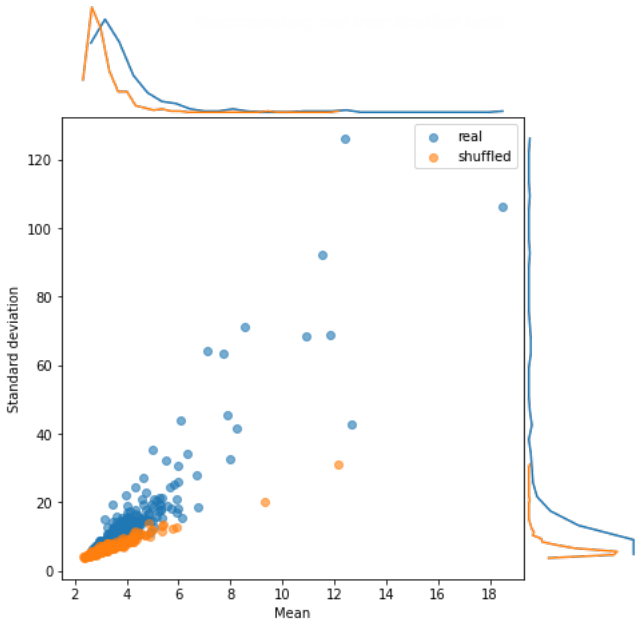
To further sustain this observation, a second assessment is performed: create a chart where x axis stores the RS mean and y axis stores the RS standard deviation, and plot each one of the 600 datapoints in the established plane. Figure 22 shows this generated plot, with the original texts plotted in bluse and their meaningless versions in orange. It is visually noticeable how the two groups are linearly separable in the plot, which is endorsed by the distribution density plots at the top and on the right of the chart. Therefore, it is possible to confirm the hypotheses that the described methodology is, to some extent, able to discriminate narratives with an actual contextual semantic from meaningless texts.

6.3 Literary genre discrimination

There are several distinct genres in the literature. They are important because they can establish patterns and associations between different texts by placing them in the same category. The idea of the experiment proposed in this section is to leverage this concept and check if the recurrence network captures a sufficient amount of contextual information from a text to be able to predict to which literary genre that text fits into.

The following two sections will describe the work that was done around that problem. The first one will explain the dataset modelling that was employed in order to find specific literary labels for the set of books in question. The second will describe the experiment of using these gathered labels in order to verify the validity of the methodology proposed in order to discriminate between them. The Recurrence Signature was also employed in an attempt to discriminate between literary genres. However, none of the experiment employed yielded sufficiently relevant results, so they will not be discussed further in this document.

Figure 22 – Scatter plot of dataset books where x is the mean of the RS and y is its standard deviation. Colored according to the text version, real or shuffled. Generated using the *matplotlib* Python library.



Source: Elaborated by the author.

6.3.1 Communities analysis

As described in Section 5.1, one complexity point of the dataset is the lack of a specific genre definition for a given book. This can be an issue when trying to work with supervised learning, since there is not an established set of labels. Henceforth, a parallel study of the Gutenberg subjects set was proposed, and its developments will be explained in this section.

Firstly a bipartite network was constructed between all books in the Gutenberg dataset and their listed subjects, linking a book with every one of its literary genres correspondents. Next, a projection onto the genres was created from the bipartite graph. Each node in the projected network corresponds one of the subject labels provided by Gutenberg. A connection between two subjects is established if there is at least one book in the dataset that contains both of their labels in its subjects set. Consequently, a network with approximately 40 thousand nodes was obtained, containing every genre that is mentioned in the Gutenberg Project. Its visualization is shown in Figure 23.

By pure observation, one can see that there are three main communities in the constructed network. In order to understand these communities and attempting to simplify the set of labels, the Louvain community detection algorithm (BLONDEL *et al.*, 2008) was ran into the network, the network nodes into three separate groups. The resulting communities are colored in the plot for visualization: first one, in orange, marked as “Fiction"; the second, in blue, marked as “Juvenile fiction"; and the third, in green, marked as "Other". These label names were chosen after an empirical examination of the data points in each of the communities. For the first two

Figure 23 – Visualization of the projection network onto genres for the whole Gutenberg dataset, colored by the community detected. Generated using the software by (SILVA, 2015).

Source: Elaborated by the author.

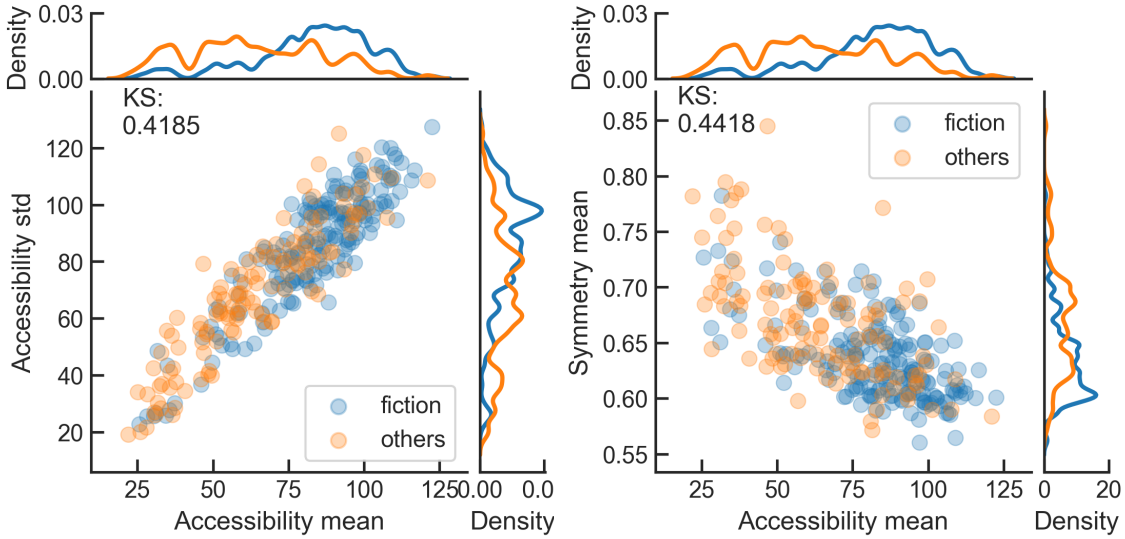
communities, the most common literary genres found were novels, fairy tales, romances, etc, with the main difference between them being the target audience: adult for the first and juvenile for the second. As for the third community, it comprised a much broader set of genres, including most of the other known categories: historical books, biographies, science books, etc, which is strongly compatible to the sparseness found for this community.

Finally, these communities can be referred to in order to define a label for a given book. For that, consider any book and its set of subjects: one can find to which community each of the genres belong to and, after, what is the most common community for that set of subjects. This will be the literary label assigned to that specific book: *fiction* (which also comprises juvenile fiction) and *others*.

6.3.2 Using accessibility and symmetry to discriminate between literary genres

Now that the dataset is labeled, it was possible to have a ground truth comparison to evaluate the performance of the proposed methodology in discriminating between literary genres. For that, we referred again to the three topological measures mentioned in Section 6.2.1: accessibility mean ($mean_\alpha$), accessibility standard deviation (std_α) and symmetry mean ($mean_S$). Similar to Figure 22, Figure 24 displays the result obtained for this experiments, the main difference being that the data points are now colored by their literary labels: blue for “fiction" and orange for “others".

Figure 24 – Discriminating literary genres using accessibility's mean and standard deviation extracted from the recurrence network, on the left, with KS = 0.4185. On the right, the same discrimination task, but now using accessibility and backbone symmetry means, with KS = 0.4418.



Source: Elaborated by the author.

Nevertheless, the resulting plots are considerably different from the ones observed in the previous experiment. At first appearance, there is no trivial separation between the two groups for neither of the plots. However, at a more careful analysis, one can see that the two density distribution plots are slightly different for each of the groups, specially for the plot on the right, between accessibility mean and symmetry mean. This perception is corroborated by the KS score obtained, which is sufficiently high to state that there is at least a discrete separation between the two distributions.

This experiment is particularly interesting, because it indicates how the recurrence network methodology is able to, to some extent, differentiate between literary genres. This assumption also comes with the fact that there must be some differences in the writing style between those two categories. These similarities within the same communities and differences across them, would be perceived by the methodology, while it extracts characteristics from the contextual narrative in the text.

6.4 Comparison between the proposed methodology and other orthodox approaches

Throughout this work, the main present goal has been to elaborate on a valid, innovative method to model texts as networks in order to characterize it through its semantic context. That being said, all the experiments considered thus far had the main objective of corroborating to the hypothesis that the intuitive idea behind modelling recurrence networks and extracting

measures to define their topological features was enough to successfully capture a narrative's semantic structure. However, it is recognizable how it can be important to compare the proposed methodology to others from the literature, even as a form of having a comparison ground to the problems being studied. Therefore, two extra experiments are proposed for this analysis, not in order to check if the recurrence network methodology can outperform any of the other approaches, but to verify how well the proposed method can perform in comparison to others more orthodox designs from the literature. The following two sections will summarize these two assessments.

6.4.1 Comparison with co-occurrence network

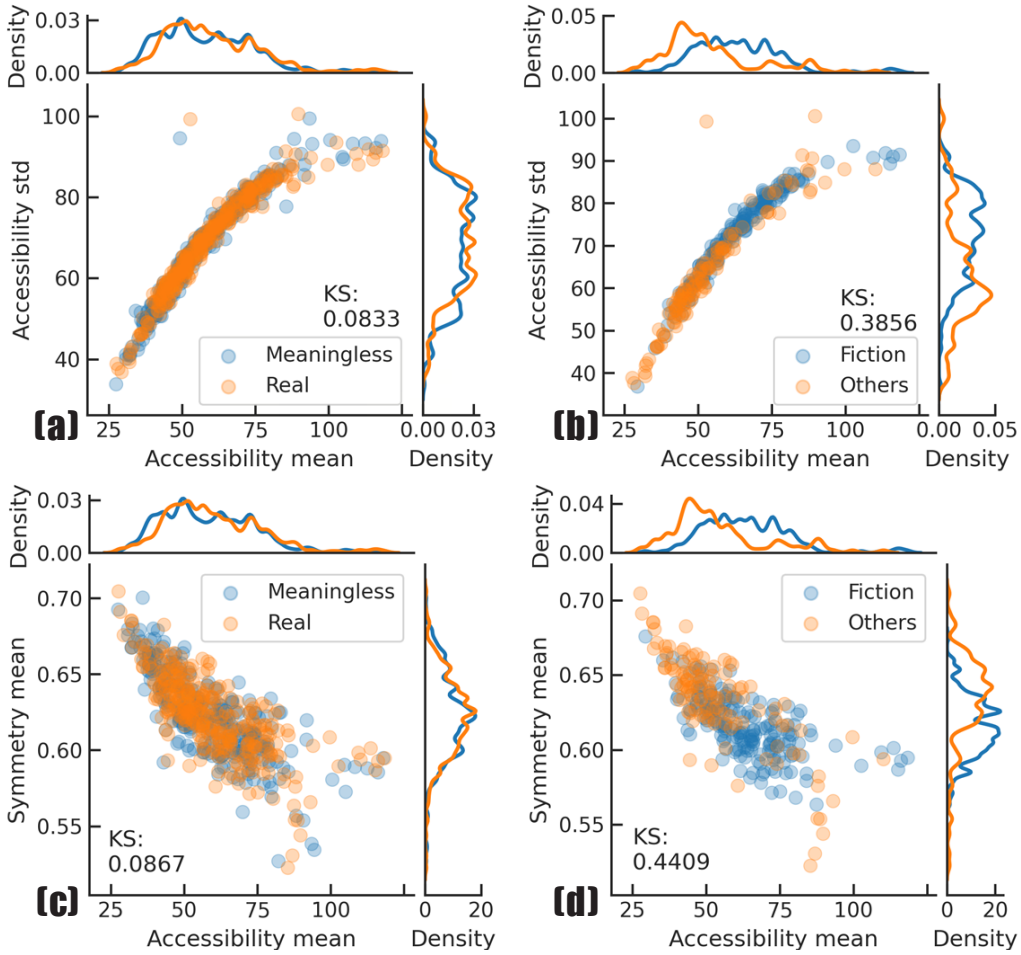
As it was discussed in Chapter 2, co-occurrence networks are a widely use methodology to model texts as networks. In this type of modelling, each term is connected to its adjacents, which reflects the linear sequence of events in a text from a micro perspective. That being said, the idea of the experiment proposed in this section is to compare the performance of using co-occurrence networks against recurrence networks, in solving the two proposed experiments thus far: discriminating between real and meaningless texts and differentiating literary genres. This test is of particular interest since it puts in trial the different perspectives of both modelling options: the recurrence network from a mesoscopic point of view against the microscopic perception of the word adjacency network.

One of the aspects that had to be carefully taken into consideration when modelling the dataset into co-occurrence networks, was the fact that the books had a great variety of different sizes. When comparing a book that has a number of words that is an order of magnitude greater than another's, it would not be fair to compare the extracted measurements for each of them since the network sizes would be so dissimilar. Therefore, it was necessary to truncate all texts in the dataset, so that they would all have the same number of words in the vocabulary. The threshold chosen here was 8306, since it was the size of the smallest vocabulary on the dataset. Finally, with this reduced version of the texts, it was possible to perform the same tests as before, by using the accessibility and symmetry measurements extracted from the co-occurrence networks modelled.

The results of these tests are illustrated on Figure 25. On the left, on plots (a) and (c), are the results yielded from using the co-occurrence approach for the problem of discriminating between real and meaningless texts, while, on the right (plots (b) and (d)), the literary genre discrimination. On the top row, on plots (a) and (b), axis x and y represent, respectively, accessibility mean ($mean_\alpha$) and accessibility standard deviation (std_α). While, on the bottom row (plots (c) and (d)), they represent accessibility mean ($mean_\alpha$) and symmetry mean ($mean_S$), respectively.

It is possible to visually notice how the recurrence network performs poorly on both tasks, more evidently on the discrimination between real and meaningless texts, with a KS of less than 0.1 for both tries. This result could be trivially predicted by the design modelling of this problem.

Figure 25 – Plots illustrating how the co-occurrence network modelling performs on assessing the proposed experiments. On the left, discriminating between real and meaningless texts. On the right, distinguishing between textual genres.



Source: Elaborated by the author.

Since only the paragraphs order was randomized for the generation of the meaningless set of texts, the adjacent words within every paragraph was preserved, excluding only the peripheral ones. Given that the number of “inside words" in a paragraph is usually a lot higher than two (the peripheral ones), the network yielded for the original version of a book would be almost completely identical to the one from the randomized version, making it practically impossible to differentiate between them. Nonetheless, this test is still important to sustain the fact that this is not a trivial problem, which enlighten the achievement of the proposed recurrence networks' satisfying performance in this task.

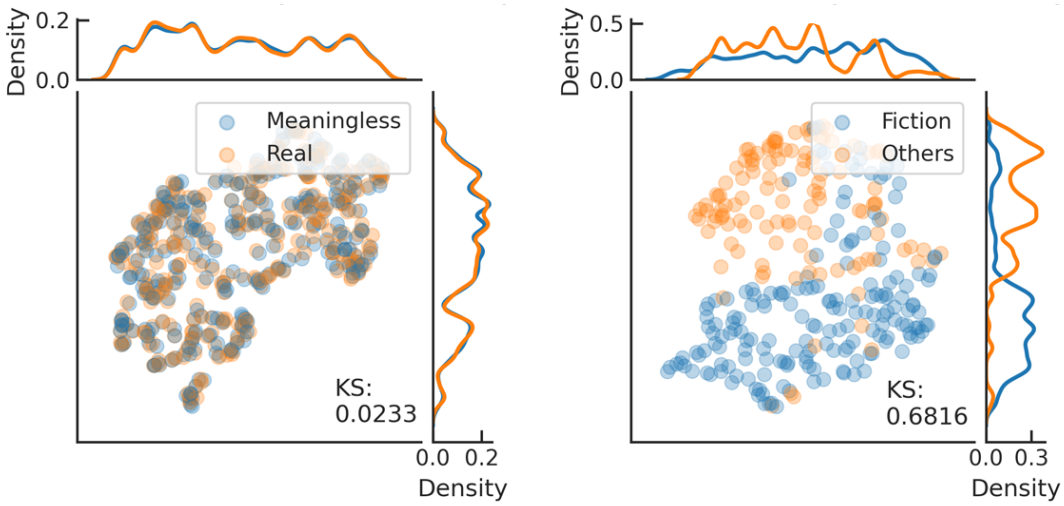
The same goes for the problem of discriminating between literary genres. From plots (b) and (d) one can tell how the co-occurrence network did not have an outstanding performance in solving the task. Obtaining the highest value of $KS = 0.4409$, which is slightly less than the score obtained from the recurrence network for the same experiment configuration, it is possible to state that both network modellings had a similar average performance on this task. Again, this is a satisfactory turnout for the methodology proposed in this work, which proves to successfully

capture, at least to some extent, the semantic properties of texts.

6.4.2 Comparison with a doc2vec modelling

Another technique from the literature that was put into test against the recurrence network was the doc2vec modelling (LE; MIKOLOV, 2014). For this experiment, every text in the dataset was transformed into its respective doc2vec representation by using the Python Gensim library (ŘEHŮŘEK; SOJKA, 2010). The vector size empirically chosen for modelling was equal to 128 and the number of epochs for learning was established as 40, in order to guarantee the methods' convergence. All the other parameters were chosen as the default values from the library. Finally, in order to represent the obtained result in a 2-dimensional plane, the UMAP technique (MCINNES; HEALY; MELVILLE, 2018) was employed to reduce the dimensionality of the vectors generated. The results of this experiment are depicted in Figure 26.

Figure 26 – Plots illustrating how the doc2vec representation performs on assessing the proposed experiments. On the left, discriminating between meaningful and meaningless texts with distinct approaches. On the right, distinguishing between textual genres. The plot is constructed after applying the UMAP dimensionality reduction technique.



Source: Elaborated by the author.

Similar to what happened for the co-occurrence approach when addressing the problem of discriminating between real and meaningless texts, the doc2vec method yields some very poor results. An obtained KS score of 0.0233 indicated that the this approach completely fails in perceiving this difference between the texts. Once more, it is important to notice that this result was also expected by the problem design, since the doc2vec method extracts information from the text as a whole, not as a sequential stream of information. In other words, it does not take into account a sequential order of paragraphs when extracting the text's patterns, so it is expected that it would not be able to differentiate between those two categories of texts.

On the other hand, the plot on the right of Figure 26 portraits a fairly satisfactory result. There is a clear visual separation between the clusters shown in the plot, the "fiction" category

Table 3 – KS values obtained for different representations: recurrence networks, co-occurrence networks and doc2vec. In bold are highlighted the best-obtained result for each of the two assessed tasks.

	Recurrence		Co-occurrence		Doc2vec
	$(mean_\alpha, std_\alpha)$	$(mean_\alpha, mean_S)$	$(mean_\alpha, std_\alpha)$	$(mean_\alpha, mean_S)$	
Real & Meaningless	0.7717	**0.8517**	0.0833	0.0867	0.0233
Fiction & Others	0.4185	0.4418	0.3856	0.4409	**0.6816**

positioned on the bottom portion of the plane while the “others" category takes the top portion of the space. This can also be noticed by observing the density distribution plot on the right of the plot, which is clearly different for both categories. This result is also sustained by the high value obtained for the KS score, equals to 0.6818. In summary, one can state that the doc2vec approach had a successful performance in differentiating between the literary categories proposed, outperforming both the recurrence and co-occurrence methods.

6.4.3 Comparison summary

On Table 3, it is shown a summary of the results from the aforementioned experiments for comparison. The KS values obtained are shown for the two discrimination problems, between real and meaningless and between literary genres, for the three approaches tested: recurrence networks, co-occurrence networks and doc2vec. For the first two, two values are shown, one for each of the attempts tested: the first using the pair $(mean_\alpha, std_\alpha)$ and the second using $(mean_\alpha, mean_S)$. The top KS value for both experiments are highlighted in bold.

Regarding the first experiment, differentiating between real and meaningless texts, it is possible to notice how the recurrence network approach was the only one that yielded a fairly positive outcome, having the higher KS score of approximately 0.85 for the approach using $(mean_\alpha, mean_S)$. On the other hand, the other two methods had poor performances in this task, all yielding KS scores of less than 0.1. Once again, this result indicates how the recurrence network modelling is indeed capable of grasping some sort of overall contextual meaning from a narrative, since it is the only methodology tested that could successfully discriminate between texts that clearly have a chronological sense of narrative from others that do not.

In contrast, for the assessment of discriminating between literary genres, the doc2vec approach had the best performance, with a $KS = 0.68$. As for the other two approaches, they both generated similar KS scores, all around 0.4, which can be considered a fair-to-middling result. That being the case, another simple test was proposed in order to compare the outcome for both network modellings: calculating the correlation between the measures calculated against each network modelling. Regarding $mean_\alpha$, the values yielded were $\rho = 0.129$ and $r_s = 0.184$. For std_α, the results were $\rho = 0.087$ and $r_s = 0.115$. And finally, for $mean_S$, $\rho = 0.284$ and $r_s = 0.274$. Since none of the correlation values obtained were significant enough, it is possible to say that, even though both approaches yielded similar results, at least the information captured by them is different, which is also an interesting outcome.

CHAPTER

7

CONCLUSIONS

In this Chapter, the final remarks of the work developed in this Masters is outlined. It is presented a review of the results, contributions and suggestions for possible future works and improvements. Given the current scenario of an increasing demand for more sophisticated methods to tackle text processing problems, numerous new approaches emerged in the literature. Many of them consider different aspects of the text, such as syntax, semantics, relationship between elements, and others. However, while the majority of them focus on a micro scale, they fail to fully capture the overall story flow of a narrative.

Inserted in this context, this Masters' work proposes an innovative technique to tackle the text representation problem, in order to address the current general limitations in the literature regarding extracting the overall contextual meaning of a text's content. Here, it is considered the syntactical dependency of the text and the network modelling is approached from a mesoscopic scale through the use of recurrence networks. Ultimately, it is wanted to exhibit how it is possible to represent the semantics of a narrative while incorporating the story flow of a book into a mesoscopic perspective, syntactical based, recurrence network modelling.

Several interesting problems and tests were performed while focusing on the mesoscopic characterization of texts. For that, we pursued an investigation of to what extent two types of characterization approaches (topological measures accessibility symmetry and the proposed recurrence signature), when calculated from the recurrence network modelling that was proposed in the present work, can discriminate between textual properties, including the presence of a real narrative and different literary genres. Finally, to evaluate how well the methodology was able to perform at the given tasks, the results were compared to the ones yielded from other, orthodox methods in the literature, namely co-occurrence networks and doc2vec modeling.

7.1 Contributions

The dataset considered comprised 300 books from the Gutenberg Project, organized into two major genres after a careful study of the subject labels provided in the Gutenberg metadata. Initially, the texts were pre-processed, the co-reference resolution technique was applied and a syntactic analysis to select terms that had the most contextual meaning in a sentence, namely verb, subject and object. After the text was split into paragraphs and a sequence of paragraphs was assigned to each of the nodes in the recurrence network, the edges were established by considering the cosine similarity between each vertex.

After this modelling, the method was validated by running four different experiments, that were intended to assess the particularities, strengths and limitations of the method put on test. For the first one, the recurrence signature proposed is compared to a set of trivial network measures. Additionally, it is compared to the other two important measures for this research: accessibility and symmetry. We find that there is not a significant enough correlation between any pair of measures to invalidate their potential validity on extracting important network structure properties.

For the second experiment, the method's performance on discriminating between real and meaningless texts is assessed. By the visual plots yielded and the KS scores obtained, both for the approaches using the recurrence signature and using the topological measures, one can conclude that the two groups were successfully separated apart. This can strongly indicate that the methodology used is indeed capturing, at least to some extent, the story flow and the contextual meaning of a narrative, in the cases where there is one.

Similar to the second test, the third experiment aimed to discuss how well the proposed method would perform in discriminating between literary genres. Even though the separation between the two labels was not visually trivial, a fair to fair-to-middling KS score obtained of 0.44 is not negligible, and still advocates in favor of the topological measures extracted from the recurrence signature having some capability of differentiating between a book's literary genre.

Finally, in the last experiment, the proposed methodology of using recurrence networks is compared to the usage of orthodox co-occurrence networks and, additionally, a doc2vec approach. By comparing the KS score values obtained, it is noticeable that the recurrence network approach was the only one that yielded a good performance on the task of discriminating between real and meaningless. On the other hand, when differentiating literary genres, the doc2vec had a considerably superior performance, while both the recurrence and the co-occurrence approaches yielded a fair-to-middling result. However, even though the obtained KS scores for the latter two were similar, a final test of correlation (using both Pearson and Spearman values) advocates that the the two approaches are not significantly correlated and, therefore, each capture distinct textual properties.

7.2 Limitations and future work

Due to resource limitations, we acknowledge the limitations of the dataset used in this research. Given that it was only considered a set of 300 books that were openly available through Project Gutenberg and, more specifically, the limitations regarding the literary genre classification issue previously discussed in this document, it would be particularly interesting to take into consideration additional books and genres in any future work. Having a larger and more diverse dataset would collaborate and emphasize any potential result yielded by the genre discrimination experiments.

In addition to having a more robust dataset, it would also be desired to consider other complementary measures including some BERT-related approaches (DEVLIN *et al.*, 2018; HAN *et al.*, 2022). In particular, methods that are able to capture the semantic aspects of a narrative. This, in order to perform a deeper evaluation of our method's ability to grasp the semantic context and the narrative story flow of texts. The recurrence approach could even be adapted to other types of texts that do not necerrarly have a narrative structure, such as lyrics (PATRA; DAS; BANDYOPADHYAY, 2017).

It would also be specially interesting to evaluate if the proposed network representation could be used for more practical applications. For instance, using it to improve the quality of recommendation systems fed by the features extracted from the recurrence networks. Another possible application is the science of science scope. In that case, nodes could represent scientific papers. The researchers from each paper would be connected by similarity and by adjacency in order to model the dynamics of their research interests over time.

7.3 Publications

The main contributions of this Master's research are reported in the following publication:

- e Souza, B. C.; SILVA, F. N.; de Arruda, H. F.; da Silva, G. D.; COSTA, L. da F.; AMANCIO, D. R. **Text characterization based on recurrence networks**. Information Sciences, v. 641, p. 119124, 2023. ISSN 0020-0255.

BIBLIOGRAPHY

ALSUMAIT, L.; BARBARÁ, D.; DOMENICONI, C. On-line lda: Adaptive topic models for mining text streams with applications to topic detection and tracking. In: **2008 Eighth IEEE International Conference on Data Mining**. [S.l.: s.n.], 2008. p. 3–12. Citation on page 27.

ALTMANN, E. G.; PIERREHUMBERT, J. B.; MOTTER, A. E. Beyond word frequency: Bursts, lulls, and scaling in the temporal distributions of words. **PLoS ONE**, Public Library of Science (PLoS), v. 4, n. 11, p. e7678, Nov 2009. ISSN 1932-6203. Available: <http://dx.doi.org/10.1371/journal.pone.0007678>. Citation on page 27.

AMANCIO, D. A complex network approach to stylometry. **PloS one**, v. 10, 06 2015. Citations on pages 28 and 41.

AMANCIO, D. R. Authorship recognition via fluctuation analysis of network topology and word intermittency. **Journal of Statistical Mechanics: Theory and Experiment**, IOP Publishing, v. 2015, n. 3, p. P03005, Mar 2015. ISSN 1742-5468. Available: <http://dx.doi.org/10.1088/1742-5468/2015/03/P03005>. Citations on pages 28 and 41.

AMANCIO, D. R.; NUNES, M. G.; OLIVEIRA, O. N.; COSTA, L. da F. Extractive summarization using complex networks and syntactic dependency. **Physica A: Statistical Mechanics and its Applications**, v. 391, n. 4, p. 1855–1864, 2012. ISSN 0378-4371. Available: <https://www.sciencedirect.com/science/article/pii/S0378437111007953>. Citations on pages 25, 28, and 48.

AMANCIO, D. R.; SILVA, F. N.; da F. Costa, L. Concentric network symmetry grasps authors' styles in word adjacency networks. **EPL (Europhysics Letters)**, IOP Publishing, v. 110, n. 6, p. 68001, 2015. Citation on page 57.

ARRUDA, H.; COSTA, L. da F.; AMANCIO, D. Using complex networks for text classification: Discriminating informative and imaginative documents. **EPL (Europhysics Letters)**, v. 113, p. 28007, 01 2016. Citations on pages 25, 28, 41, and 47.

ARRUDA, H. F. de; SILVA, F. N.; MARINHO, V. Q.; AMANCIO, D. R.; COSTA, L. da F. Mesoscopic representation of texts as complex networks. **CoRR**, abs/1606.09636, 2016. Available: <http://arxiv.org/abs/1606.09636>. Citations on pages 25, 28, 49, 55, 57, and 63.

BARABASI, A.-L.; ALBERT, R. Emergence of scaling in random networks. **Science**, v. 286, n. 5439, p. 509–512, 1999. Available: <http://www.sciencemag.org/cgi/content/abstract/286/5439/509>. Citation on page 33.

BLONDEL, V.; GUILLAUME, J.-L.; LAMBIOTTE, R.; LEFEBVRE, E. Fast unfolding of communities in large networks. **Journal of Statistical Mechanics Theory and Experiment**, v. 2008, 04 2008. Citation on page 67.

BOCCALETTI, S.; LATORA, V.; MORENO, Y.; CHAVEZ, M.; HWANG, D.-U. Complex networks: Structure and dynamics. **Physics Reports**, v. 424, p. 175–308, 02 2006. Citation on page 31.

BREDE, M.; NEWTH, D. Patterns in syntactic dependency networks from authored and randomised texts. **Complexity International**, v. 12, 10 2008. Citation on page 27.

BRISCOE, T. Introduction to linguistics for natural language processing. In: . [S.l.: s.n.], 2014. Citations on pages 27 and 41.

BUNT, H.; BOS, J.; PULMAN, S. **Computing Meaning**. [S.l.]: Springer Publishing Company, Incorporated, 2013. ISBN 9400772831. Citations on pages 27 and 43.

COSTA, L. da F.; ANDRADE, R. F. S. What are the best concentric descriptors for complex networks? **New Journal of Physics**, IOP Publishing, v. 9, n. 9, p. 311–311, sep 2007. Available: <https://doi.org/10.1088/1367-2630/9/9/311>. Citation on page 37.

DEVLIN, J.; CHANG, M.-W.; LEE, K.; TOUTANOVA, K. Bert: Pre-training of deep bidirectional transformers for language understanding. **arXiv preprint arXiv:1810.04805**, 2018. Citation on page 77.

DONNER, R. V.; ZOU, Y.; DONGES, J. F.; MARWAN, N.; KURTHS, J. Recurrence networks—a novel paradigm for nonlinear time series analysis. **New Journal of Physics**, IOP Publishing, v. 12, n. 3, p. 033025, 2010. Citations on pages 28 and 29.

ERDÖS, P.; RÉNYI, A. On random graphs i. **Publicationes Mathematicae Debrecen**, v. 6, p. 290, 1959. Citation on page 32.

ESTRADA, E. **The Structure of Complex Networks: Theory and Applications**. USA: Oxford University Press, Inc., 2011. ISBN 019959175X. Citations on pages 28, 34, and 41.

FELDMAN, R.; SANGER, J. **The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data**. [S.l.]: Cambridge University Press, 2006. Citation on page 43.

FRUCHTERMAN, T. M. J.; REINGOLD, E. M. Graph drawing by force-directed placement. **Software: Practice and Experience**, v. 21, n. 11, p. 1129–1164, 1991. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1002/spe.4380211102>. Citation on page 50.

HAN, S.; SHI, L.; RICHIE, R.; TSUI, F. R. Building siamese attention-augmented recurrent convolutional neural networks for document similarity scoring. **Information Sciences**, Elsevier, v. 615, p. 90–102, 2022. Citation on page 77.

INDURKHYA, N.; DAMERAU, F. J. **Handbook of Natural Language Processing**. 2nd. ed. [S.l.]: Chapman & Hall/CRC, 2010. ISBN 1420085921, 9781420085921. Citation on page 53.

LE, Q.; MIKOLOV, T. Distributed representations of sentences and documents. In: PMLR. **International conference on machine learning**. [S.l.], 2014. p. 1188–1196. Citations on pages 45 and 72.

LEITE, D.; RINO, L.; PARDO, T.; NUNES, M.; NUNES, V. Extractive automatic summarization: Does more linguistic knowledge make a difference? 01 2007. Citations on pages 28 and 54.

LIN, C.-Y.; HOVY, E. Automatic evaluation of summaries using n-gram co-occurrence statistics. In: **Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1**. USA: Association for Computational Linguistics, 2003. (NAACL '03), p. 71–78. Available: <https://doi.org/10.3115/1073445.1073465>. Citation on page 48.

LIU, S.; WU, Y.; WEI, E.; LIU, M.; LIU, Y. Storyflow: Tracking the evolution of stories. **IEEE Transactions on Visualization and Computer Graphics**, IEEE Educational Activities Department, USA, v. 19, n. 12, p. 2436–2445, Dec. 2013. ISSN 1077-2626. Available: <https://doi.org/10.1109/TVCG.2013.196>. Citations on pages 25, 27, and 49.

MANNING, C. D.; SCHÜTZE, H. **Foundations of Statistical Natural Language Processing**. Cambridge, MA, USA: MIT Press, 1999. ISBN 0262133601. Citations on pages 42 and 54.

MANNING, C. D.; SURDEANU, M.; BAUER, J.; FINKEL, J. R.; BETHARD, S.; MCCLOSKY, D. The stanford corenlp natural language processing toolkit. In: **Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations**. [S.l.: s.n.], 2014. p. 55–60. Citations on pages 42 and 54.

MASSEY, F. J. The Kolmogorov-Smirnov test for goodness of fit. **Journal of the American Statistical Association**, American Statistical Association, v. 46, n. 253, p. 68–78, 1951. Citation on page 65.

MCINNES, L.; HEALY, J.; MELVILLE, J. **UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction**. [S.l.]: arXiv, 2018. Citation on page 72.

MEHRI, A.; DAROONEH, A.; SHARIATI, A. The complex networks approach for authorship attribution of books. **Physica A: Statistical Mechanics and its Applications**, v. 391, p. 2429–2437, 04 2012. Citation on page 41.

MIKOLOV, T.; CHEN, K.; CORRADO, G.; DEAN, J. Efficient estimation of word representations in vector space. **Proceedings of Workshop at ICLR**, v. 2013, 01 2013. Citation on page 27.

____. Efficient estimation of word representations in vector space. **arXiv preprint arXiv:1301.3781**, 2013. Citation on page 45.

NEWMAN, M. E. J. **Networks: an introduction**. Oxford; New York: Oxford University Press, 2010. ISBN 9780199206650 0199206651. Available: <http://www.amazon.com/Networks-An-Introduction-Mark-Newman/dp/0199206651/ref=sr_1_5?ie=UTF8&qid=1352896678&sr=8-5&keywords=complex+networks>. Citations on pages 28 and 31.

PATRA, B. G.; DAS, D.; BANDYOPADHYAY, S. Retrieving similar lyrics for music recommendation system. In: **Proceedings of the 14th International Conference on Natural Language Processing (ICON-2017)**. [S.l.: s.n.], 2017. p. 290–297. Citation on page 77.

PROJECT Gutenberg. 1971. Available: <https://www.gutenberg.org>. Citation on page 51.

ŘEHŮŘEK, R.; SOJKA, P. Software framework for topic modelling with large corpora. In: **Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks**. Valletta, Malta: ELRA, 2010. p. 45–50. Citation on page 72.

SEGARRA, S.; EISEN, M.; RIBEIRO, A. Authorship attribution through function word adjacency networks. **IEEE Transactions on Signal Processing**, Institute of Electrical and Electronics Engineers (IEEE), v. 63, n. 20, p. 5464–5478, Oct 2015. ISSN 1941-0476. Available: <http://dx.doi.org/10.1109/TSP.2015.2451111>. Citation on page 41.

SILVA, F. N. **Dimensão e simetria em redes complexas: uma abordagem multi-escala**. 2015. Citations on pages 14 and 68.

SILVA, F. N.; COMIN, C. H.; PERON, T. K.; RODRIGUES, F. A.; YE, C.; WILSON, R. C.; HANCOCK, E. R.; da F. Costa, L. Concentric network symmetry. **Information Sciences**, v. 333, p. 61–80, 2016. ISSN 0020-0255. Available: <https://www.sciencedirect.com/science/article/pii/S0020025515008245>. Citations on pages 37 and 63.

SPACY Python library. 2016. Available: <https://spacy.io/>. Citation on page 44.

STEELS, L. Language as a complex adaptive system. In: . [S.l.: s.n.], 2002. ISBN 978-3-540-41056-0. Citation on page 28.

TRAVENÇOLO, B.; COSTA, L. da F. Accessibility in complex networks. **Physics Letters A**, v. 373, p. 89–95, 12 2008. Citations on pages 38 and 63.

WATTS, D. J.; STROGATZ, S. H. Collective dynamics of 'small-world' networks. **Nature**, v. 393, n. 6684, p. 440–442, 1998. Citation on page 32.


ICMC
USP
SÃO CARLOS