

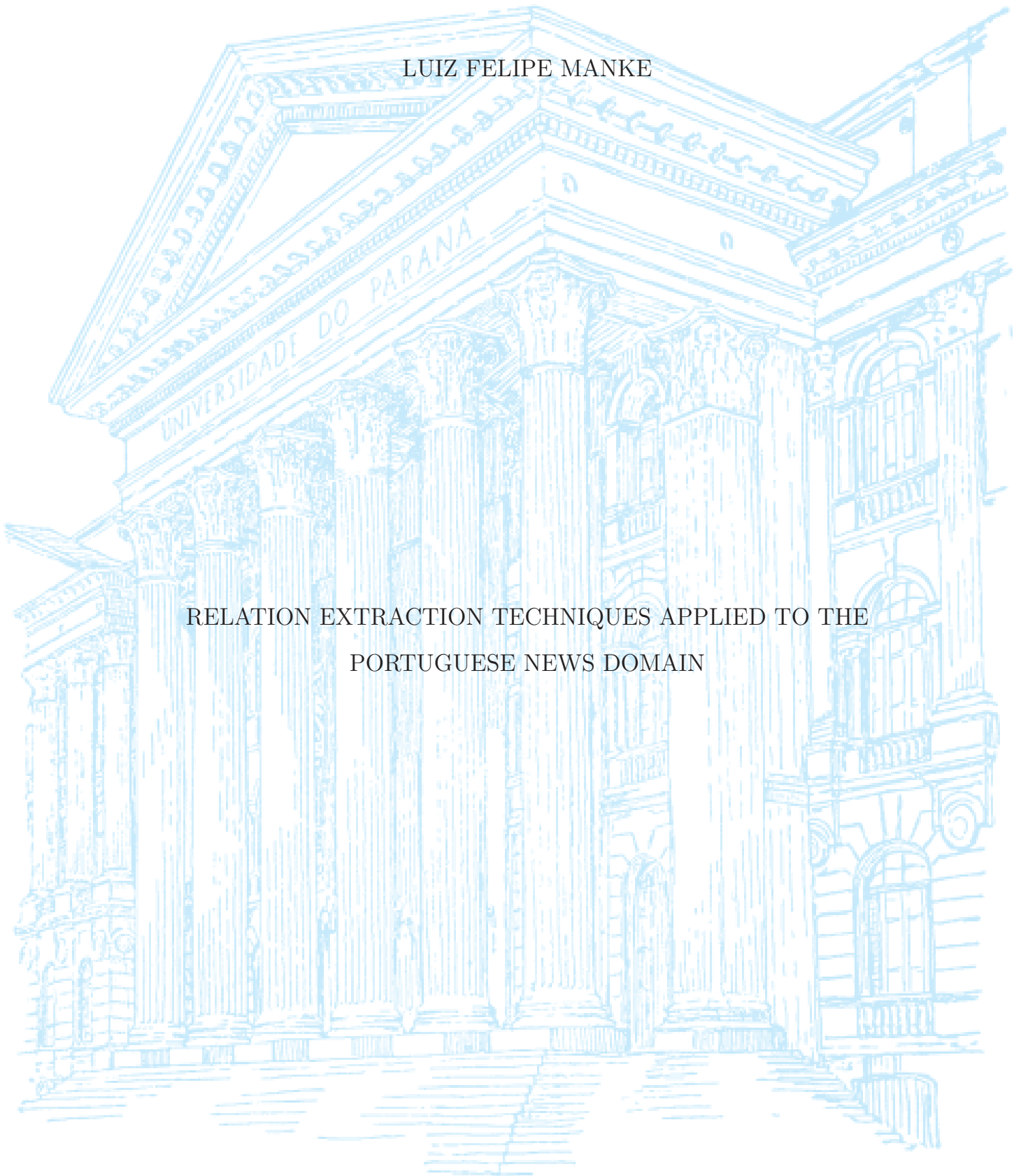
UNIVERSIDADE FEDERAL DO PARANÁ

LUIZ FELIPE MANKE

RELATION EXTRACTION TECHNIQUES APPLIED TO THE
PORTUGUESE NEWS DOMAIN

CURITIBA

2022



LUIZ FELIPE MANKE

RELATION EXTRACTION TECHNIQUES APPLIED TO THE
PORTUGUESE NEWS DOMAIN

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica, área de concentração Sistemas Eletrônicos, do Setor de Tecnologia da Universidade Federal do Paraná, como requisito parcial para a obtenção do grau de Mestre em Engenharia Elétrica.

Orientador: Prof. Dr. Leandro dos Santos Coelho

CURITIBA

2022

DADOS INTERNACIONAIS DE CATALOGAÇÃO NA PUBLICAÇÃO (CIP)
UNIVERSIDADE FEDERAL DO PARANÁ
SISTEMA DE BIBLIOTECAS – BIBLIOTECA CIÊNCIA E TECNOLOGIA

Manke, Luiz Felipe.

Relation extraction techniques applied to the Portuguese News
Domain. / Luiz Felipe Manke. – Curitiba, 2022.

1 recurso on-line : PDF.

Dissertação (Mestrado) – Universidade Federal do Paraná, Setor de
Tecnologia, Programa de Pós-Graduação em Engenharia Elétrica.

Orientador: Prof. Dr. Leandro dos Santos Coelho.

1. Engenharia elétrica. 2. Redes neurais (Computadores). 3. Notícias
- Agências. 4. Língua Portuguesa. I. Coelho, Leandro dos Santos. II.
Universidade Federal do Paraná. Programa de Pós-Graduação em
Engenharia Elétrica. III. Título.

Bibliotecário: Nilson Carlos Vieira Júnior CRB-9/1797

TERMO DE APROVAÇÃO

Os membros da Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação ENGENHARIA ELÉTRICA da Universidade Federal do Paraná foram convocados para realizar a arguição da Dissertação de Mestrado de **LUIZ FELIPE MANKE** intitulada: **Relation Extraction Techniques Applied to the Portuguese News Domain**, sob orientação do Prof. Dr. LEANDRO DOS SANTOS COELHO, que após terem inquirido o aluno e realizada a avaliação do trabalho, são de parecer pela sua APROVAÇÃO no rito de defesa.

A outorga do título de mestre está sujeita à homologação pelo colegiado, ao atendimento de todas as indicações e correções solicitadas pela banca e ao pleno atendimento das demandas regimentais do Programa de Pós-Graduação.

CURITIBA, 25 de Agosto de 2022.

Assinatura Eletrônica

14/09/2022 17:29:24.0

LEANDRO DOS SANTOS COELHO
Presidente da Banca Examinadora

Assinatura Eletrônica

13/09/2022 13:35:16.0

GIDEON VILLAR LEANDRO
Avaliador Interno (UNIVERSIDADE FEDERAL DO PARANÁ)

Assinatura Eletrônica

13/09/2022 17:20:41.0

ROBERTO ZANETTI FREIRE
Avaliador Externo (PONTIFICA UNIVERSIDADE CATÓLICA DO
PARANÁ)

Assinatura Eletrônica

13/09/2022 13:35:18.0

ALEXANDRE RASI AOKI
Avaliador Interno (UNIVERSIDADE FEDERAL DO PARANÁ)

RESUMO

Notícias negativas são definidas como qualquer tipo de informação desfavorável encontrada em uma grande variedade de fontes de notícias. A triagem dessas notícias significa identificar e monitorar uma grande variedade de conteúdos de mídias convencionais impressas e televisivas, juntamente com conteúdos on-line, como blogs e feeds de mídias sociais. Este processo deve ser uma parte importante do processo de due diligence de qualquer instituição financeira para mitigar os riscos ao iniciar qualquer relacionamento comercial. A automação é mais rápida do que a triagem humana e pode ser melhor adaptada às preocupações comerciais exclusivas de uma empresa. Com as vantagens do Machine Learning, agora é possível analisar as notícias e isolar as entidades reais que foram identificadas como tendo informações adversas sem verificar manualmente cada artigo. Estas técnicas fazem parte da área de Extração de Relacionamento, que vem do campo de Processamento de Linguagem Natural, e visa identificar automaticamente as associações semânticas (relações) entre as entidades nomeadas no texto. Esta dissertação propõe desenvolver diferentes soluções de Extração de Relacionamento para a tarefa específica de automatizar o processo de busca de notícias negativas dentro da língua portuguesa. Com base em um estudo cuidadoso da estrutura teórica e uma revisão sistemática da literatura sobre o assunto, três abordagens diferentes foram propostas para resolver o problema: (i) um método de Embedding Sub-sequences, que usa os vetores de incorporação das entidades como entrada para um classificador de Floresta Aleatória; (ii) uma Graph Convolutional Neural Network, que usa vetores de incorporação junto com árvores parciais de dependência como entrada para camadas convolutivas; e (iii) uma Transformer Neural Network, que usa uma rede neural pré-treinada que se baseia exclusivamente em mecanismos de atenção. Para avaliar as metodologias propostas, foi construído um benchmark em português para o domínio notícias, onde os textos em nível de sentença das notícias foram coletados manualmente, processados e rotulados em uma categoria binária. Os métodos foram comparados em termos da pontuação F1, que é a média harmônica entre as métricas de precisão e recall. Após uma validação cruzada de cinco dobras, a metodologia Transformer mostrou resultados notavelmente melhores do que as outras abordagens, atingindo uma pontuação de F1 de 91,72, que foi 10% maior do que a pontuação do Graph CNN, e 27% maior do que a pontuação do Embedding Sub-sequences. Estes valores também são significativos quando comparados ao estado da arte. Além disso, foram aplicados os métodos Shapley Additive Explanations para melhor compreender a influência que cada característica de entrada (palavras) tem no valor previsto, o que também destacou como o método Transformer superou as outras metodologias, dando grande atenção a palavras importantes próximas às entidades, enquanto as outras metodologias não puderam se concentrar no contexto principal das frases.

Palavras-chave: árvores de dependência; redes neurais; notícias; língua portuguesa; extração de relação; representação de palavras.

ABSTRACT

Negative news is defined as any kind of unfavorable information found across a wide variety of news sources. Screening for these news means identifying and monitoring a large range of conventional printed and televised media content, along with online content like blogs and social media feeds. This process should be an important part of any financial institution's customer due diligence process to mitigate risks when starting any business relationship. Automation is faster than human screening and can be better tailored to a company's unique business concerns. With the advantages of Machine Learning, now it's possible to analyze news and isolate the actual entities that have been identified as having adverse information without manually checking each article. These techniques are part of the Relation Extraction area, which comes from the Natural Language Processing field, and aims to automatically identify semantic associations (relations) between named entities in text. This thesis proposes to develop different Relation Extraction solutions to the specific task of automating the process of negative news search within the Portuguese language. Based on a careful study of the theoretical framework and a systematic literature review on the subject, three different approaches were proposed to solve the problem: (i) an Embedding Sub-sequences method, that uses the embedding vectors from around the entities as input to a Random Forest classifier; (ii) a Graph Convolutional Neural Network, that uses embedding vectors along with dependency parse trees as input to convolutional layers; and (iii) a Transformer Neural Network, which uses a pre-trained neural network that is based solely on attention mechanisms. To evaluate the proposed methodologies, a Portuguese benchmark was built for the News domain, where sentence-level texts from news were manually collected, processed, and labeled into a binary category. The methods were compared in terms of the F1 score, which is the harmonic mean between precision and recall metrics. After a 5-fold cross-validation, the Transformer methodology showed notably better results than the other approaches, reaching an F1 score of 91.72, which was 10% higher than the Graph CNN score, and 27% higher than the Embedding Sub-sequences score. These values are also significant when compared to state-of-the-art results. Additionally, Shapley Additive Explanations were applied the methods to better understand the influence that each input feature (words) has on the predicted value, which also highlighted how the Transformer method overcame the other methodologies by giving great attention to important words close to the entities, while the other methodologies couldn't focus on the main context of the sentences.

Keywords: dependency trees; neural networks; news domain; portuguese language; relation extraction; word embedding.

LIST OF ILLUSTRATIONS

Figure 1.1 – Project methodology.	11
Figure 2.1 – Relation extraction settings.	13
Figure 2.2 – Relation extraction methods.	14
Figure 2.3 – Example of a pattern-oriented method.	15
Figure 2.4 – CNN, RNN, and self-attention architectures.	16
Figure 2.5 – Example of a tree-parsed sentence.	18
Figure 2.6 – Word embedding example.	21
Figure 3.1 – Annual distribution of relation extraction articles.	23
Figure 3.2 – Most common keywords for Relation Extraction in Scopus database. . .	26
Figure 4.1 – Embedding Sub-sequences methodology.	33
Figure 4.2 – Graph Convolutional Neural Network methodology.	34
Figure 4.3 – Graph Convolutions.	35
Figure 4.4 – Transformers architecture.	36
Figure 4.5 – Transformers methodology.	37
Figure 4.6 – Dataset methodology.	39
Figure 4.7 – Confusion matrix.	41
Figure 4.8 – Shapley Additive Explanations.	42
Figure 5.1 – Embedding Sub-sequences Confusion Matrix.	46
Figure 5.2 – Embedding Sub-sequences Probability Plot.	47
Figure 5.3 – Embedding Sub-sequences Negative Samples Shapley Values.	47
Figure 5.4 – Embedding Sub-sequences Positive Samples Shapley Values.	48
Figure 5.5 – Graph Convolutional Neural Network Confusion Matrix.	50
Figure 5.6 – Graph Convolutional Neural Network Probability Plot.	51
Figure 5.7 – Graph Convolutional Neural Network Negative Samples Shapley Values. .	52
Figure 5.8 – Graph Convolutional Neural Network Positive Samples Shapley Values. .	52
Figure 5.9 – Transformer Confusion Matrix.	54
Figure 5.10–Transformer Probability Plot.	55
Figure 5.11–Transformer Negative Samples Shapley Values.	56
Figure 5.12–Transformer Positive Samples Shapley Values.	56

LIST OF TABLES

Table 3.1 – Most frequent research areas.	24
Table 3.2 – Most frequent source titles.	25
Table 5.1 – Embedding Sub-sequences Scores.	45
Table 5.2 – Graph Convolutional Neural Network Scores.	50
Table 5.3 – Transformer Neural Network Scores.	54

LIST OF ABBREVIATIONS AND ACRONYMS

ACE	Automated Content Extraction
BERT	Bidirectional Encoder Representations from Transformers
CBOW	Continuous Bag Of Words
CNN	Convolutional Neural Network
GCNN	Graph Convolutional Neural Network
GPU	Graphics Processing Unit
IDF	Inverse Document Frequency
IE	Information Extraction
KDE	Kernel Density Estimation
LSA	Latent Semantic Analysis
LSTM	Long Short-term Memory
NER	Named Entity Recognition
NLP	Natural Language Processing
NN	Neural Network
PCA	Principal Component Analysis
POS	Part-Of-Speech
RE	Relation Extraction
RNN	Recurrent Neural Network
SVD	Singular Value Decomposition
SHAP	Shapley Additive Explanations
TF	Term Frequency

CONTENTS

1	Introduction	9
1.1	Project Methodology	10
1.2	Document Structure	11
2	Theory Framework	12
2.1	Relation Extraction	12
2.1.1	Tasks	12
2.1.2	Approaches	13
2.2	Supervised Learning	15
2.2.1	Pattern-oriented	15
2.2.2	Sequence-oriented	16
2.2.3	Tree-oriented	17
2.2.4	Graph-oriented	18
2.3	Word Embedding	19
2.3.1	Count-based	19
2.3.2	Prediction-based	20
2.4	Final Chapter Considerations	21
3	Literature Review	23
3.1	Selection process	23
3.2	Summary of the most cited works	25
3.2.1	Feature-based methods	27
3.2.2	Dependency parse methods	27
3.2.3	Convolutional neural network methods	28
3.2.4	Sequence-based methods	29
3.2.5	Portuguese domain	30
3.3	Final Chapter Considerations	31
4	Methodology	32
4.1	Methods	32
4.1.1	Embedding Sub-sequences	32
4.1.2	Graph Convolutional Neural Network	33
4.1.3	Transformer Neural Network	35
4.2	Dataset	38

4.2.1	News Sources	38
4.2.2	Preparation	38
4.2.3	Data Labeling	39
4.3	Evaluation Metrics	40
4.3.1	F1 score	40
4.3.2	Shapley Additive Explanations	42
4.4	Final Chapter Considerations	43
5	Application	44
5.1	Embedding Sub-sequences	44
5.2	Graph Convolutional Neural Network	48
5.3	Transformer Neural Network	53
5.4	Final Chapter Considerations	57
6	Final Considerations	58
6.1	Future Works	59
	REFERENCES	60

1 Introduction

News is the part of communication that keeps people informed of the changing events, issues, and characters in the world outside. It may be provided through many different media: word of mouth, printing, postal systems, broadcasting, electronic communication, or through the testimony of observers and witnesses to events.

Negative news is defined as any kind of unfavorable information found across a wide variety of news sources. Identifying this news should be an important part of any financial institution’s customer due diligence process. When breaking news reveals a client’s involvement with unethical or criminal activity, firms need to be able to identify and assess these stories as efficiently as possible in order to reassess this client’s risk profile and protect themselves from reputational damage. Between 2018-2019 alone, inadequate compliance and screening controls have cost top banking institutions and other non-financial firms millions of dollars in fines (KHANDPUR et al., 2021).

Screening for negative news means identifying and monitoring a wide range of conventional printed and televised media content, along with online content like blogs and social media feeds. Since the process traditionally involves manual checks and human interpretation, the sheer amount of information firms has to deal with during screening makes it labor-intensive and time-consuming. To manage the information collected, best practice guidance suggests using automation to complement the manual elements of the review process.

Automation is obviously faster than human screening and can be better tailored to a company’s unique business concerns – but it also has pitfalls. Setting screening parameters too indiscriminately may result in an overwhelming amount of information for human reviewers, and therefore the same inefficiency problems while screening too narrowly increases the chances of missing an important story. As a result, financial institutions of all sizes frequently see false positive rates in excess of 90 percent, and substantial manual investigation time is allocated to ultimately benign activity (AGGARWAL; WAREHAM; LEHMANN, 2020; HAN et al., 2018).

With the advantages of Machine Learning, now it’s possible to analyze news and isolate the actual entities that have been identified as having adverse information without manually checking each article. More specifically, this area of research is called Natural Language Processing (NLP) and includes applications such as sentiment analysis, text classification, machine translation, text summarization, and so on.

Within the NLP field, Relation Extraction (RE) is an important area that aims to automatically identify semantic associations (relations) between named entities in text. This topic has been receiving increasing attention due to the exponential growth of the

information made available on the web, which makes RE tasks impossible through manual means.

In order to extract this information from text, people have been working on automatic Relation Extraction techniques for years. From early pattern matching rules to current neural networks, existing RE methods have achieved significant progress in several areas, such as question answering systems, text summarization, information retrieval, and the construction of lexical resources and ontologies.

Several approaches have been proposed for RE from unstructured data, such as supervised or unsupervised learning; corpus-based techniques, linguistic strategies; resources such as lexical bases and ontologies; rule-based models, models based on statistical learning, and hybrid systems. For some languages, such as English, there is extensive research on RE, while for Portuguese, we find a smaller number of works dealing with relation extraction.

1.1 Project Methodology

To better plan the development of this thesis, five steps were defined, as presented in Figure 1.1.

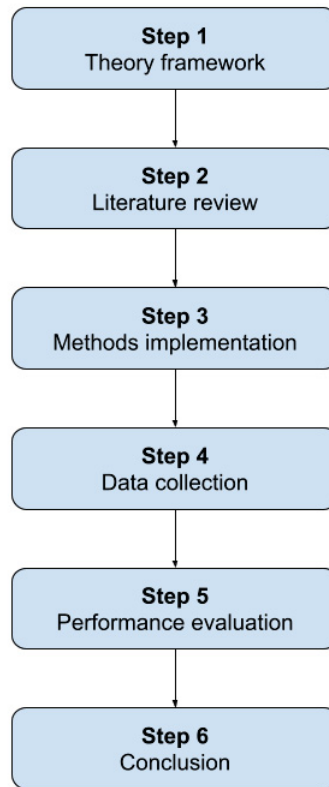
The first step is the theory framework. The goal here is to understand most of the ways Relation Extraction is applied, including different tasks and applications, the techniques used on the problems, and the different tools used to represent text to a machine learning model.

Next, once there's a general understanding of the problems and techniques that RE is used for, the literature review aims to search for the most relevant works published so far. This step includes finding the most successful approaches, the benchmarks normally used, as well as the metrics and results achieved on these papers.

Step 3 is the implementation of the proposed methodologies, which should be based on the knowledge obtained in the previous steps. Since this thesis focuses on the specific domain of Portuguese news, step 4 proposes the data collection and labeling to form a new benchmark.

Finally, step 5 is the evaluation of each proposed methodologies on the benchmark and metrics selected. They should be compared not only with each other but also with the state-of-the-art scores found during the literature review process. The conclusion of the findings will be presented in step 6.

Figure 1.1 – Project methodology.



Source: The author (2022).

1.2 Document Structure

This document is organized into six chapters. Chapter 1 introduced the problem that will be addressed, presented the goals, and detailed the research methodology. In Chapter 2, the main theory behind most of the Relation Extraction techniques is carefully organized, including RE tasks and approaches, supervised machine learning algorithms, and word embedding methods. Chapter 3 presents the methodology used to perform a systematic literature reviews and a brief explanation of the most cited works found. Chapter 4 introduces the selected state-of-the-art methods that will be further implemented, along with the evaluation metrics and the benchmark created. In Chapter 5, the application and results for each of the RE methods are exposed. Finally, Chapter 6 presents the discussion of the results and the conclusion of the thesis.

2 Theory Framework

This chapter aims to present an overview of Relation Extraction. First, it's provided a perspective of various tasks and categories that RE can be associated with, depending on the information one aims to extract, according to Christopoulou (2020). Then, a few methods developed to tackle RE tasks are presented, elaborating on their advantages and limitations. Finally, the remainder of the chapter discusses approaches to encoding text so that computers can understand and process it (SANDHYA, 2019).

2.1 Relation Extraction

Relation Extraction is a part of the Natural Language Processing field that aims to automatically identify semantic associations (relations) between named entities in text. It can be divided into multiple tasks that focus on particular types of relations, formulated in different settings, and different approaches developed to handle these tasks. This section will cover both the tasks and approaches used in RE.

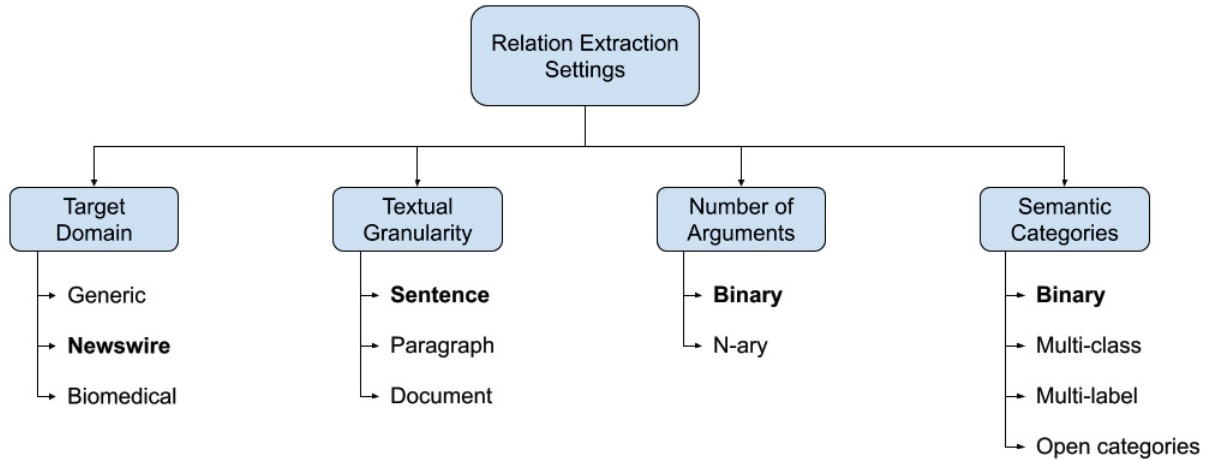
2.1.1 Tasks

For RE tasks, Figure 2.1 shows a diagram of what will be discussed in this subsection. An initial distinction can be made based on the target domain. Approaches for relation extraction were initially applied to generic data, extracted from the web or encyclopedias such as Wikipedia. However, multiple methods have been adapted or developed for a particular domain. For instance, due to the rapid growth of the total bibliographical data available, the biomedical domain has gained a lot of attention to extract structured information, such as protein-protein interactions, from scientific literature and clinical reports.

The next more general category refers to the textual granularity from which one aims to extract relations between named entities. Sentence-level RE, i.e. extraction of relations within the same sentence, is the most well-studied RE task. However, some studies focus on inter-sentence RE, i.e. extraction of relations beyond sentences, which deals with several complex semantic relationships in documents, requiring local, non-local, syntactic, and semantic dependencies.

Relation tasks can also be separated concerning the number of arguments participating in the relation. In the case of two entities being involved in a relation, the task is defined as binary RE, which is the most commonly-studied RE setting. Any relation between more than two named entities falls in the category of n-ary RE. The type of

Figure 2.1 – Relation extraction settings.



Source: The author (2022).

arguments between which one aims to extract a relation can also be considered a different RE task. Mention-level RE indicates extraction between named entity mentions, while concept-level RE refers to the extraction of relations between concepts.

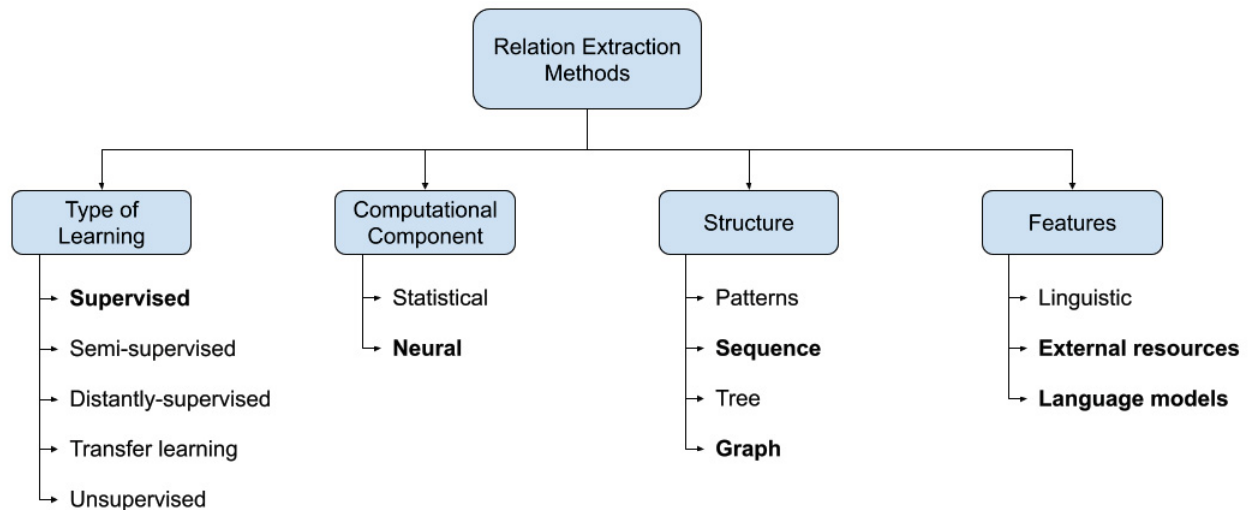
Another RE categorization is defined based on the semantic categories that named entities can share. When relation extraction is treated as a classification problem, relations are typically chosen from a set of pre-defined relation categories. The simplest categorization is binary classification, involving only two possible relation categories, i.e. related and non-related. In the case of a larger set of relation categories, e.g. located-in, founder-of, etc., the classification is named multi-class. Finally, if a given group of entities can be assigned more than one potential relation type simultaneously, the classification is known as multi-label. However, when relation extraction is treated as an Open Information Extraction (IE) setting, relation categories are extracted automatically from text, without pre-defined sets.

2.1.2 Approaches

As shown in Figure 2.2, the first RE approach is based on the type of learning. Here, learning is defined as the method used inside the model to exploit the provided textual data, either raw, human, or automatically annotated. This category can be subdivided into different learning approaches, including learning only from annotated data (supervised), learning by leveraging additional unlabeled data (semi-supervised), learning using knowledge from external resources (distantly-supervised), methods leveraging data from rich domains for prediction in lower resource domains (transfer learning), as well as

methods that do not require any amount of labeled data (unsupervised).

Figure 2.2 – Relation extraction methods.



Source: The author (2022).

The second category is based on the computation component that each approach uses, which can be statistical or neural approaches. The former typically makes use of explicit features to represent relation instances and feed them into machine learning classifiers, kernel functions, or probabilistic graphical models to track dependencies between random variables. While the latter takes advantage of recent advances in neural networks to represent instances as dense vectors with a fixed size, namely embeddings, to achieving state-of-the-art results.

Another division considers the type of underlying structural representation, where existing approaches can be categorized into pattern-, sequence-, tree- and graph-oriented. These representations can be either implicitly (in the form of features) or explicitly (by transforming the input into certain structures). The next section will dig deeper into each one separately, although it's possible to employ a combination of different structures.

Finally, we can divide methods based on the type of features they incorporate. Typical features include linguistic features, features obtained from external resources such as Knowledge Bases or semantic inventories, as well as automatically constructed features obtained from language models, primarily used by neural architectures. For instance, networks such as Word2Vec (MIKOLOV et al., 2013) and Glove (PENNINGTON; SOCHER; MANNING, 2014) automatically learn rich word representations from raw textual sources.

2.2 Supervised Learning

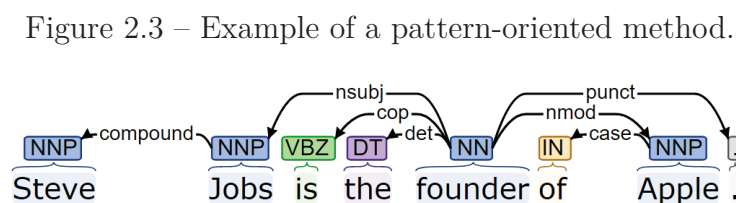
Although there are several types of techniques to use in Relation Extraction, this thesis focuses only on supervised techniques, where the goal is to classify a set of arguments into a particular set of pre-defined relation categories.

The timeline of RE supervised approaches begins with feature-based methods. It requires that the input instances are transformed into fixed-size feature vectors - something that needs a careful selection of features. Later, kernel-based methods were employed to represent instances in a latent space and compute a score of similarity between them to increase the expressiveness by investigating implicit feature spaces. This approach has the advantage of computing the dot product between two high-dimensional vectors without explicitly mapping it into the new space, but still requires a set of manually generated features as input. More recently, neural models came to replace previous approaches using rich representations, instead of hand-crafted features, to represent words.

In this section, we provide a brief explanation of such techniques grouped by their structural representation. This means that it won't follow a strict timeline, but instead, it aims to show that older structures are consistently used with more sophisticated computational components, such as Neural Networks.

2.2.1 Pattern-oriented

Patterns were the initial form of structure used to extract relations from text. It can use not only exact word matches to find relations but also information coming from external tools, such as part-of-speech (POS) tags and dependency parses. For instance, Figure 2.3 shows a sentence with its tags and dependencies. From this sentence, one could consider the pattern “*is the founder of*” to find similar relations between *Steve Jobs* and *Apple*.



Source: The author (2022).

For instance, Ferreira, Oliveira & Rodrigues (2019) extracts relations based on POS rules bootstrapped from positive training examples. An interesting approach taken

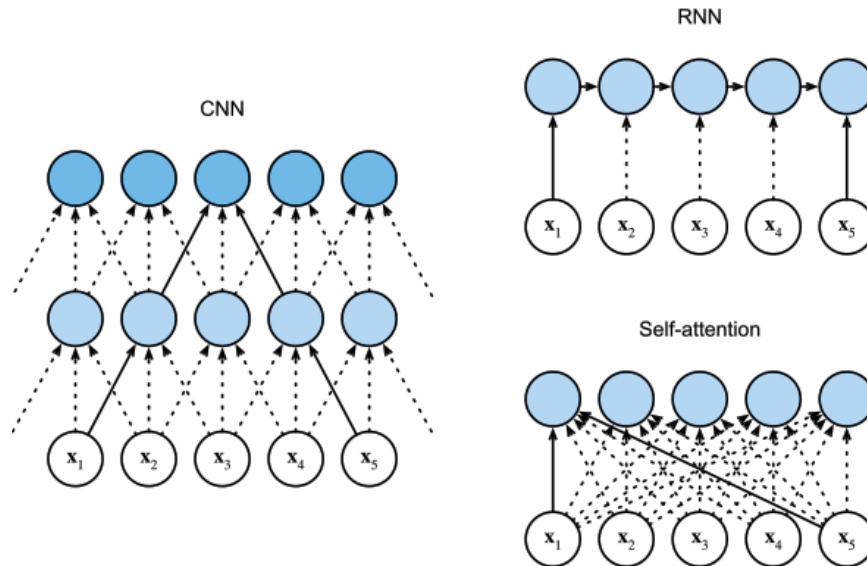
to generate more rules was to remove some rules constraints by making smaller rules out of the original ones. This approach achieved an F1 score of 72.0 within the HAREM collection.

Such approaches, however, were quickly abandoned, particularly with the creation of human-annotated datasets, since they were too restrictive in identifying multiple expressions that describe the same thing. Instead, methods moved on to representing text with other forms of structure such as sequences, trees, or graphs, which use automatically generated patterns.

2.2.2 Sequence-oriented

The goal of sequence-based methods is to represent a set of arguments given their corresponding context as a sequence, considering word order. The rise of sequence-based methods was coincidental with the introduction of neural networks, especially networks that were developed to perform on sequences or sub-sequences. Figure 2.4 presents the three approaches that will be discussed in this subsection.

Figure 2.4 – CNN, RNN, and self-attention architectures.



Source: Zhang et al. (2020).

The first such models were Convolutional Neural Networks (CNNs), which emerged from their extensive usage in image processing. For RE tasks, CNNs use a combination of vectors as word representations, such as pre-trained word embeddings, and positional features as input to identify the entities.

Despite the effectiveness that CNNs showed as sequential models for relation extraction, they had the disadvantage of not being able to model long-range dependencies well, as they focused on encoding local context using a short window. This was something that could be solved by Recurrent Neural Networks (RNNs) and their variants, as neural models developed particularly for sequences. The Long-Short Term Memory (LSTM) Network is a particular variant that showed to be effective in modeling long sequences.

Attention-based approaches were proposed along with Recurrent Networks as well. It tries to overcome the information bottleneck of the intermediary state caused by long sequence modeling by allowing the decoder model to access all the hidden states, rather than a single vector, to predict each output. Transformers are Neural Networks (NNs) architectures that are based solely on these mechanisms, which can be used in three different ways: (i) encoder-decoder attention, which is between input and output sequences; (ii) self-attention in the input sequence, where each position in the encoder can attend to all positions in the previous layer of the encoder; and (iii) self-attention in the output sequence, which allows positions of the decoder to attend to all positions in the previous layer of the decoder up to and including that position (to prevent information leakage).

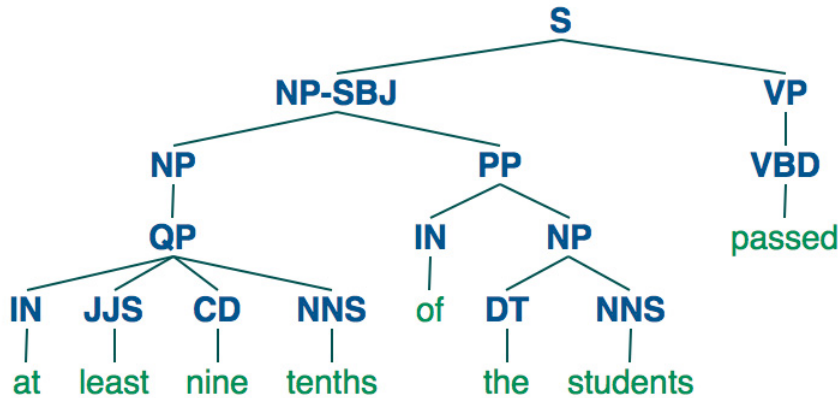
Nguyen & Grishman (2015) make use of a Convolutional Neural Network with position embeddings for encoding relative distances between words to beat state-of-the-art models on baselines. It also uses multiple window sizes for filters to automatically learn contextualized features from the sentence. Soares et al. (2019) propose a novel training setup, which relies solely on entity resolution annotations and it's based on the Transformers architecture.

2.2.3 Tree-oriented

Trees were one of the first representations used (after rules and patterns) to represent language and, compared to sequence-oriented approaches, are more complex in terms of structure. This approach relies on external syntactic and dependency parsers that can provide tree-base structures for any textual snippet. It can either use the dependency-based information derived from parsers as features in machine learning classifiers or use a tree-oriented structure in the model directly. In both cases, the dependency on the external tools restricts the domain portability, as well as propagates potential parsing errors inside the models. Figure 2.5 illustrates these trees.

For instance, to extract *person-affiliation* and *organization-location* relations from text, Zelenko, Aone & Richardella (2003) used tree structures as input to kernel methods, which were used to compute the similarity between two sentences. Fundel, Küffner & Zimmer (2007) used dependency parse trees in combination with simple handcrafted rules to extract interactions between genes and proteins from free text. And Plank & Moschitti

Figure 2.5 – Example of a tree-parsed sentence.



Source: Bird, Klein & Loper (2009).

(2013) explores syntactic tree kernels enriched by lexical semantic similarity to solve domain adaptation problem for relation extraction systems.

2.2.4 Graph-oriented

In comparison to trees, graphs are more flexible structures that enable multiple types of interactions. A major difference between them is that the former is restricted to having only a single parent node, whereas the latter enables arbitrary connections between nodes.

Graph-oriented approaches use a graph structure and attempt to encode it via machine learning algorithms. Existing approaches can be broadly divided into two categories: methods that simply perform on existing graph structures, i.e. the input graph is given to the model; and methods that first construct the graph and then use a particular model to encode the interactions in the structure.

About the first category, pre-defined graphs are knowledge represented in a defined schema in the form of triples consisting of two arguments and a relation between them. Since relations are always extracted between named entity concepts rather than mentions, knowledge graphs contain factual relations and are not dependent on a specific sentence.

Finally, approaches that construct graph structures and then encode them, typically rely on trees obtained from a parser and utilize heuristics to transform them into graphs. The majority of such approaches have been proposed for cross-sentence relation extraction.

Bunescu & Mooney (2005) approaches relation extraction using kernel function

based on dependency graphs to get top-level relations from the ACE (Automated Content Extraction) newspaper corpus. Zhang, Qi & Manning (2018) also use dependency graphs in combination with Convolutional Neural Networks on TACRED benchmark to improve computation efficiency and information extraction. Moreover, Guo, Zhang & Lu (2019) proposes a graph CNN to selectively attend to the relevant sub-structures on dependency graphs.

2.3 Word Embedding

Since computers do not understand text, we need some kind of numeric representation of these inputs so that machines can understand and process them. This section explores some techniques used for this purpose.

2.3.1 Count-based

Count-based methods are primarily based on the technique called Latent Semantic Analysis (LSA) which uses documents as features of words based on the hypothesis that similar words would occur in similar documents. Some of the most common count-based methods are Count Vector, TF-IDF, and Co-occurrence Matrix.

Count Vector is a simple method that learns unique words from all the documents and then represents each word by counting the number of times it appears in each document. The more words are found, the larger the dimension of the vector.

TF-IDF is a frequency-based method that not only takes into account the occurrence of a word in a single document but also the whole corpus of documents. It weighs down the common words appearing in almost all documents while giving more weight to less common words. Term Frequency (TF) is the frequency of each word appearing in a document. The importance of a word increases proportionally to the number of times it appears in an individual document. Inverse Document Frequency (IDF) is a measure of how common the word is across all documents. If a word appears in almost all the documents then it might not be as important as compared to words that appear in a few documents. Equation 2.1 expresses this method, where $TFIDF_{x,y}$ is the TF-IDF for the term x over the document y , $TF_{x,y}$ is the term frequency in the document, N is the number of documents, and df_x is the number of documents containing x .

$$TFIDF_{x,y} = TF_{x,y} \times \log \left(\frac{N}{df_x} \right) \quad (2.1)$$

Finally, the Co-occurrence Matrix tries to capture the semantic relationship between words by taking into account the neighboring words in specific window sizes. The matrix is computed by counting how two or more words occur together in a given corpus.

The resultant matrix can be very sparse depending on the size of the vocabulary. If the vocabulary comprises N unique words, Co-occurrence Matrix will be of dimensions N by N . The technique can capture the powerful semantic and syntactic relationships in the vocabulary but it is computationally expensive due to high dimensionality. Therefore, techniques like Principle Component Analysis (PCA) and Singular Value Decomposition (SVD) are used to decompose Co-occurrence Matrices to reduced dimensions with the least data-loss as possible.

Although count-based methods are simple, two words with related meanings might be assigned distant integers, while two adjacent words assignment might have nothing to do with each other. So, there is no relationship in the words and information cannot be easily shared across words with similar properties. Moreover, this representation of words as unique and discrete results in data sparsity.

To overcome this problem, the next subsection will introduce a representation of words that capture their meanings, semantic relationships, and the different types of contexts they are used in.

2.3.2 Prediction-based

Predictive word representations are based on the idea that a word vector should encode its meaning using the neighboring words. Such representations of words use shallow neural networks to learn the word vectors as parameters of the model.

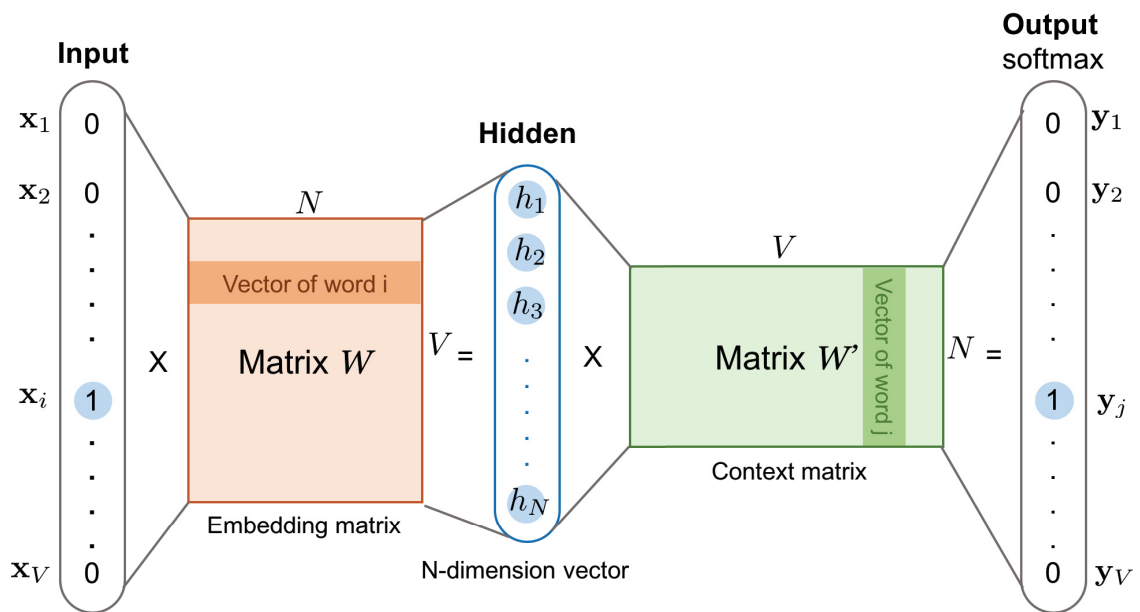
To preserve the information around a word token, one can derive a notion of vector representation rather than integer representation for words where each dimension of the vector can be used for different purposes. For example, one dimension in the vector can represent the root each word belongs to the words eat, eaten, eats, ate would be assigned 1 in the particular dimension while all other words that do not belong to this root would be assigned 0.

The idea of predictive-based word representation, also called word embeddings, has been around for some time but it was in 2013 when Mikolov, Chen, et al. came up with two neural network-based architectures to create high-dimensional word representations capturing semantic relationships between words unaided by external annotations. The Continuous Bag of Words (CBOW) is one of two model architectures introduced. CBOW model learns word representations by a probabilistic feed-forward neural network to predict the target word given the context. The other proposed model is the skip-gram model, which is similar to the CBOW model but instead of predicting the target word based on context, it tries to predict the context words within a radius given the target word. So, skip-gram architecture reverses the use of target and center words. Figure 2.6 illustrates an example of these techniques.

For example, Shen & Huang (2016) proposes an attention-based Convolutional

Neural Network for semantic relation extraction on SemEval-2010 Task 8 benchmark that takes word embeddings as inputs, where the embedding model was a CBOW trained on 100 billion words from Google News. Another CNN for drug-drug interaction was proposed by Liu et al. (2016), which uses word embeddings and position embeddings from a lookup table to capture semantic information of words and relative distances between the two drugs of interest.

Figure 2.6 – Word embedding example.



Source: Weng (2017).

2.4 Final Chapter Considerations

The previous sections presented an overview of various RE tasks and approaches, a few supervised-learning methods, and word embeddings used to enable machines to interpret human language. The goal of this current section is to highlight some specific techniques that will be applied in the rest of this document.

The target domain chosen is pretty straightforward, since the purpose of this project is to apply RE to the news domain. The text granularity selected was the sentence-level, due to the simplicity of not having co-reference between sentences. The number of arguments will be binary since the relation will be extracted between only two entities.

Lastly, there will be only two possible semantic categories as output, whether the entities contain any relation or not.

For RE methods, this thesis focus only on one type of learning, the supervised-learning, which is usually the approach that contains the most amount of resources available, as well as the highest achieved scores. The computational components used will be a combination of statistical and neural network approaches. The same goes for the structures used, being possible to combine sequence and graph neural networks, for example. Finally, the features will mainly be focused on language models, which are prediction-based word embeddings, but external tools will also be used during implementations.

3 Literature Review

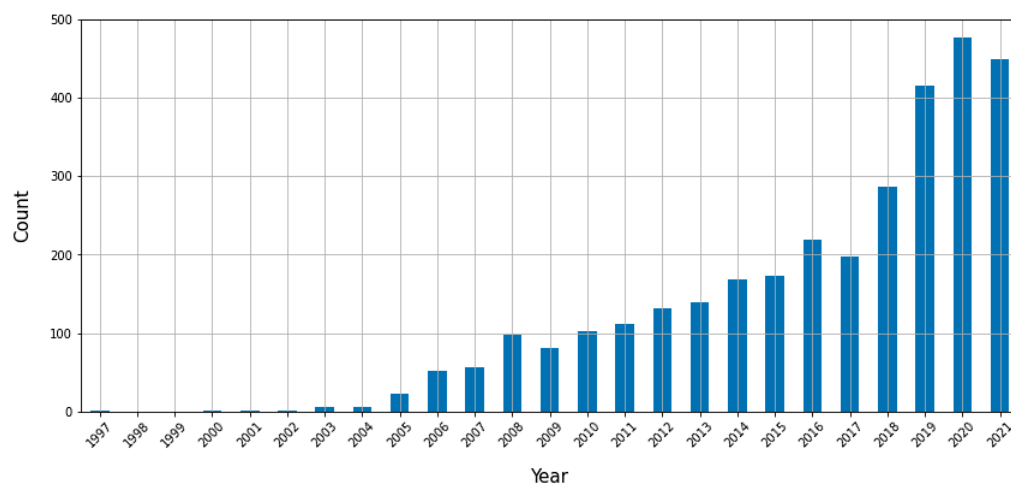
This chapter presents the search methodology, research sources, and selection criteria used in the literature review process, as well as a summary of the main works related to this document. In this process, a systematic review was used, which enables reproduction, as well as provides an objective way to find articles and studies relevant to the topic (GUSENBAUER; HADDAWAY, 2020).

3.1 Selection process

The search was carried out until December 31, 2021, from the Scopus database, which is the largest abstract and citation database of peer-reviewed literature from scientific journals, books, and conference proceedings. The query used to search within the title, abstract and keywords consisted only on the term “*relation extraction*”.

From the search methodology, 3,195 documents were found. Figure 3.1 shows the number of articles published by year. Until 2005 less than 10 annual articles were published and until 2010 less than 100 annual publications. In the last three years, there was an explosion of publications related to the topic, reaching almost 500 publications in 2020.

Figure 3.1 – Annual distribution of relation extraction articles.



Source: The author (2022).

Table 3.1 shows the most frequent research areas based on articles about Relation Extraction from the Scopus database, where one article can fit more than one category. Computer Science, Mathematics, and Engineering are expected to be in the top research areas, but names such as Social Sciences, Arts and Humanities, and Medicine also made the top list.

Table 3.1 – Most frequent research areas.

Research Area	Absolute Frequency	Percentage Frequency
Computer Science	2836	44.5%
Mathematics	825	12.9%
Engineering	555	8.7%
Social Sciences	541	8.5%
Arts and Humanities	449	7.0%
Medicine	315	4.9%
Decision Sciences	263	4.1%
Others	598	9.4%

Source: Blackburn & Domin (2006).

Table 3.2 shows the most frequent source titles based on articles related to Relation Extraction from the Scopus database. The studies are well distributed along the various types of sources, but again Computer Science stands out from the others.

To view the interconnection of the most present concepts in Scopus repository publications that include Relation Extraction problems and methodologies, Figure 3.2 shows a VOSviewer illustration, which is an information visualization software widely used to select the best keywords used by authors in their articles. From its colors, it's possible to visualize that in the early years, a lot of studies focused on knowledge management, data mining, and simpler algorithms. While, in more recent years, the technologies evolved and researchers started focusing on neural networks and deep learning for relation extraction and language models.

To select the articles relevant to this study from the searched documents, the following criteria were used:

Inclusion criteria

- Articles where relation extraction was the main task in the system.

Table 3.2 – Most frequent source titles.

Source Title	Absolute Frequency	Percentage Frequency
Lecture Notes in Computer Science	424	13.3%
Ceur Workshop Proceedings	139	4.3%
Computer and Information Science	86	2.7%
ACM International Conference	75	2.3%
Journals of Biomedical Informatics	45	1.4%
BCM Bioinformatics	43	1.3%
IEEE Access	39	1.2%
Others	2344	73.5%

Source: Blackburn & Domin (2006).

- Studies published in English or Portuguese language.

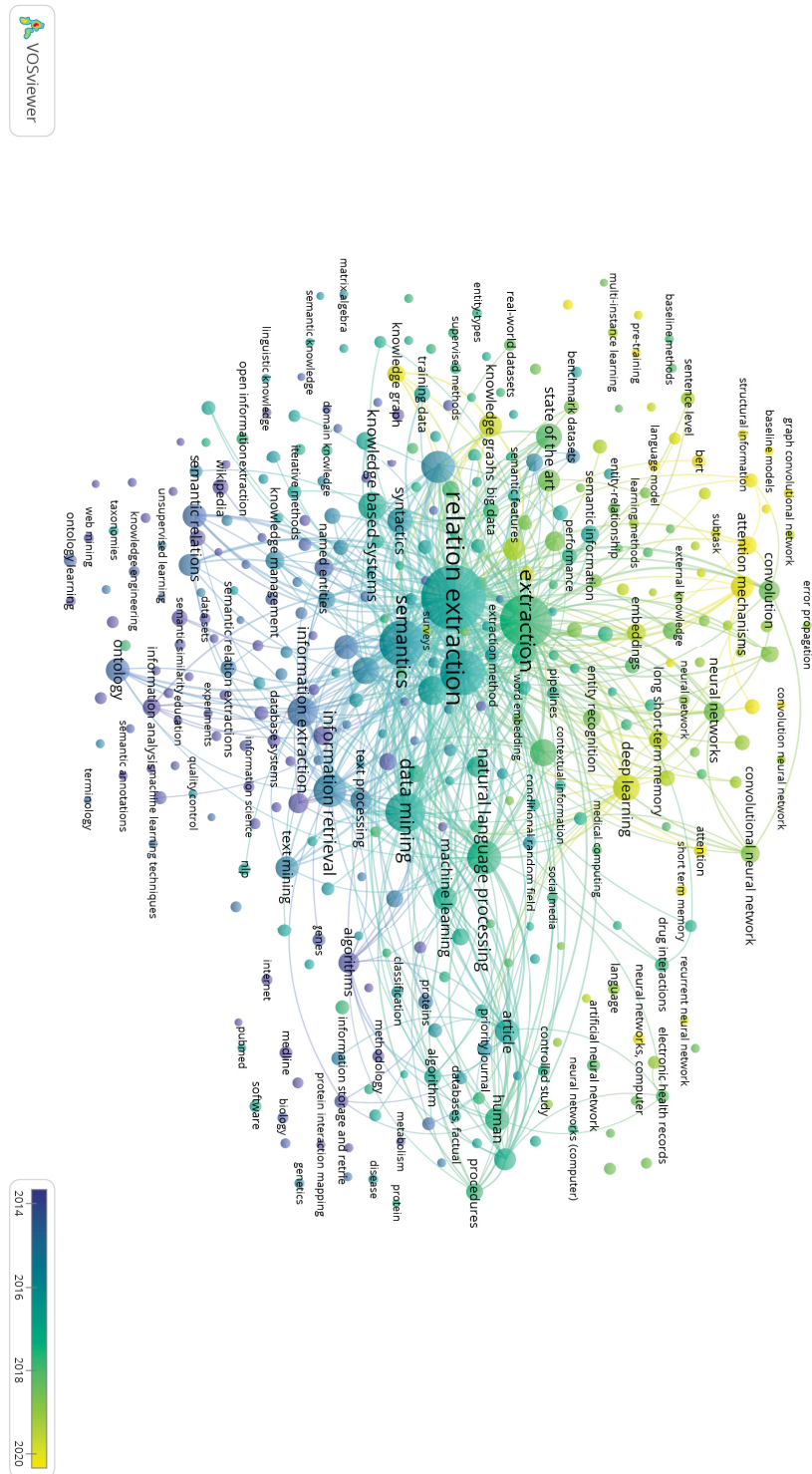
Exclusion criteria

- Papers without the query terms within the title, abstract, or keywords.
- Articles that focused on the review of methods.
- Studies that used distant or weak supervision in the relation extraction process.
- Documents where the Relation Extraction task followed a Named Entity Recognition task.
- Articles that didn't make use of Machine Learning techniques or any Natural Language Processing tool.
- Inconclusive studies or studies that had no relevant contributions to the topic of relation extraction.
- Full articles that weren't available.

3.2 Summary of the most cited works

The following subsections will describe the most cited works found in this systematic literature review. The papers are grouped by their main methods and are organized in chronological order.

Figure 3.2 – Most common keywords for Relation Extraction in Scopus database.



Source: The author (2022).

3.2.1 Feature-based methods

To improve feature-based approaches, the incorporation of diverse lexical, syntactic and semantic knowledge was explored by Zhou et al. (2005). This study illustrates that the base phrase chunking information is very effective for RE and contributes to most of the performance improvement from the syntactic aspect, while additional information from full parsing gives limited further enhancement. Experiments on the ACE corpus showed that the proposed approach significantly outperforms previous tree kernel-based methods, reaching an F1 score of 74.7 on relation detection tasks.

Abacha & Zweigenbaum (2011) also proposes a combination of various features for a supervised machine learning approach to discover relations between medical problems, treatments, and tests mentioned in electronic medical records using little knowledge mining. A single support vector machine classifier was used to identify relations between concepts and to assign their semantic type. Several resources such as Wikipedia, WordNet, General Inquirer, and a relation similarity metric were used to enrich the classifier with lexical and contextual features. This technique obtained the highest score in the 2010 i2b2 Challenge, with an F1 score of 73.7.

3.2.2 Dependency parse methods

Zelenko, Aone & Richardella (2003) proposed to tackle RE tasks using kernel-based machine learning methods. The kernels were defined over shallow parse trees representation of text, and were used in conjunction with Support Vector Machine and Voted Perceptron algorithms to extract *person-affiliation* and *organization-location* relations from text. It achieved superior performances compared to feature-based methods on a news domain corpus, reaching an F1 score of 86.8.

Based on the observation that the information required to assert a relationship between two named entities in the same sentence is typically captured by the shortest path between the two entities in the dependency graph, Bunescu & Mooney (2005) proposed a novel approach to RE, also based on kernels. This method achieved an F1 score of 52.5 on extracting top-level relations from the ACE corpus.

Another kernel method for extracting semantic relations between entities in natural language text was proposed by Mooney & Bunescu (2005). It is based on a generalization of subsequence kernels, which uses three types of patterns that are typically employed in natural language to assert relationships between entities: (i) the text before the first entity; (ii) the text between the entities; and (iii) the text after the second entity. This method showed better performance than previous rule-based and dependency tree kernels methods when applied to biomedical and newspaper corpora. The F1 score reached values up to 47.7.

Also using dependency parse trees, plus simple handcrafted rules, a tool, called RelEx, was developed by Fundel, Küffner & Zimmer (2007) to extract interactions between genes and proteins from free text. This system was applied to one million MEDLINE abstracts dealing with genes and protein relations, and it was able to extract around 150 thousand relations with an estimated performance of both 80% precision and 80% recall.

To explore the domain adaptation of Relation Extraction systems, Plank & Moschitti (2013) proposed syntactic tree kernels enriched by lexical semantic similarity. By combining word clustering, latent semantic analysis, and structured kernels this approach was able to achieve competitive performance on different ACE 2005 domains, up to a 70.1 F1 score.

3.2.3 Convolutional neural network methods

To eliminate the errors introduced by features generated using linguistic analysis tools, Nguyen & Grishman (2015) present a new approach to Relation Extraction based on a Convolutional Neural Network. The proposed framework uses pre-trained word embeddings as an initializer, and position embeddings for encoding relative distances, and takes advantage of multiple window sizes for filters to automatically learn features from sentences and minimize the dependence on external tools. The experimental results show that the system significantly outperforms state-of-the-art systems, reaching an F1 score of 61.32 on the ACE benchmark, and 82.8 on the SemEval 2010 dataset.

Another CNN for drug-drug interaction was proposed by Liu et al. (2016). It uses word embeddings and position embeddings to capture semantic information of words and relative distances between the two drugs of interest. This approach achieved an F1 score of 69.75 on the 2013 DDIE extraction challenge, outperforming state-of-the-art systems by 2.75%, where most of them are based on support vector machines with a long number of manually defined features.

Shen & Huang (2016) proposes a novel attention-based CNN for semantic relation extraction. The word-level attention mechanism is used to better determine which parts of the sentence are most influential concerning the two entities of interest, which enables the architecture to learn important features from task-specific label data. The proposed approach also makes use of word embedding, part-of-speech tags, and position embedding. Experiments on the SemEval-2010 Task 8 benchmark dataset showed that this model achieved better performances than several state-of-the-art models with minimal feature engineering. In terms of F1 score, it reached a value of 85.9.

Although dependency trees can help relation extraction models capture long-range relations between entities, this approach can either neglect crucial information by pruning the dependency trees too aggressively, or it can be computationally inefficient because it's difficult to parallelize different tree structures. To address these problems, an

extension of Graph CNN for RE was proposed by Zhang, Qi & Manning (2018), which pools information over arbitrary dependency structures efficiently in parallel, and uses a novel pruning strategy that keeps the words immediately around the shortest path between the two entities. This methodology outperforms existing sequence and dependency-based neural models on the large-scale TACRED benchmark, achieving an F1 score of 68.2.

Although dependency trees can provide rich structural information for extracting relations between entities in text, it remains a challenge to make use of relevant information while ignoring irrelevant ones. An attention-guided graph convolutional network is proposed by Guo, Zhang & Lu (2019), a novel model which directly takes full dependency trees as input and automatically learns how to selectively attend to the relevant substructures for the task. This model gives significantly better results when compared to previous approaches on various tasks, including cross-sentence n-ary RE and large-scale sentence-level RE, reaching an F1 score of 87.0, a 2 percentage point increase.

3.2.4 Sequence-based methods

Levy et al. (2017) showed that Relation Extraction tasks can be reduced to answering simple reading comprehension questions, by associating one or more natural language questions with each relation slot. One of its advantages is the possibility of doing zero-shot learning by extracting new relation types that are only specified at test-time. Experiments on a Wikipedia slot-filling task demonstrate that this approach can generalize with high accuracy on known relations, and, at a low accuracy level, do zero-shot learning on unseen relations.

The problem of representing useful relation representation directly from text is addressed by Soares et al. (2019). A novel training setup, called *matching the blanks*, is proposed. This approach relies solely on entity resolution annotations and it's based on the transformers' architecture (specifically, BERT). This methodology significantly outperforms previous works on exemplar-based relation extraction (FewRel), even without the use of any task's training data, and on supervised relation extraction benchmarks: SemEval 2010 Task 8, KBP37, and TACRED, reaching an F1 score of 89.5 on the first dataset.

Most work in Relation Extraction forms a prediction by looking at a short span of text within a single sentence containing a single entity pair mentioned. To consider interactions across mentions, and relationships expressed across sentence boundaries, a document-level approach is proposed by Verga, Strubell & McCallum (2018). This approach simultaneously predicts relationships between all mentioned pairs in a document using an efficient self-attention encoder, achieving state-of-the-art performance on Biocreative benchmarks.

3.2.5 Portuguese domain

In order to reduce the modeling effort, Fernandes, Milidiú & Rentería (2010) proposes a machine-learning method for the extraction of structured information from text. The key idea of this paper is to model the target structures as a relation over entities, which are simpler problems than the original one. The method showed strong results when compared to token classification state-of-the-art approaches within five different tasks: phrase chunking, clause identification, hedge detection, quotation extraction, and dependency parsing. On this last task, it was able to reach a significant F1 score of 88.18.

Different types of features, used as input to Relation Extraction systems, contain different kinds of linguistic knowledge: from orthographic or lexical to more complex features. To understand the impact of the different types of inputs, four kinds of linguistic features were selected and evaluated systematically by Garcia & Gamallo (2011). The experiments were performed on the Portuguese version of Wikipedia, and, despite the combination of some types of features allowing an improvement of the F1 score (achieving 77.9%), it was observed that adjusting the positive and negative ratio of the samples could lead to a higher quality classifier with just a single type of linguistic feature, based on generic lexico-syntactic.

For the Portuguese language RE, a system, named RelP, was developed by Abreu & Vieira (2017). It is an Open RE framework, which extracts any relation descriptor that describes an explicit relation between named entities in the organization domain using the Conditional Random Fields technique. RelP achieved state-of-the-art performance, where the F1 score reached 60.0 between the entities of person, organization, and place.

Another simple approach to relation extraction in the Portuguese language was proposed by Ferreira, Oliveira & Rodrigues (2019). In the paper, the authors described the application of the NLPyPort, which has a module for extracting relations based on part-of-speech rules bootstrapped from positive training examples. An interesting approach taken to generate more rules was to remove some rules constraints by making smaller rules out of the original ones. The proposed approach achieved an F1 score of 72.0 within the HAREM collection.

Reyes et al. (2021) proposed a sequence-based approach to help decision-making in organizations. It uses Bidirectional Encoder Representations from Transformers (BERT) to process sentences and entities, outputting the part of the sentence that represents the semantic relation between the entities. The framework was tested within the Portuguese language, more specifically in the financial domain, and achieved a score of 73,5 using the Jaccard similarity metric. Moreover, the paper also made available the benchmark dataset, with more than 4,500 labeled sentences from tweets and news articles.

3.3 Final Chapter Considerations

The literature review process confirmed that the area of Relation Extraction is drawing a lot of attention. Up to the year 2022, more than 3,000 articles were found across multiple research areas, where more than 1,200 of these articles were published only in the last three years.

Among the most cited works explored in this chapter, four main approaches could be identified: (i) feature-based methods, (ii) dependency parse methods; (iii) convolutional neural network methods; and (iv) sequence-based methods. Moreover, it was possible to notice some commonly used benchmarks to evaluate these methodologies, such as the ACE, SemEval 2010 Task 8, and TACRED datasets.

One aspect that every paper has in common is its evaluation metric, the F1 score. This is a metric that takes into account both precision and recall scores to balance the predictions' false positives and false negatives. However, the state-of-the-art scores vary depending on the RE task and benchmark being evaluated. For instance, with the ACE dataset, an F1 score of almost 75 could be achieved. On the other hand, with the SemEval 2010 Task 8 dataset, an F1 score of 89.5 was reached.

Within the Portuguese language domain, the RE area is also expanding, but at a more modest pace. Some of the methodologies, such as convolutional neural networks, could not be found under this domain. Also, the Portuguese language still lacks a benchmark focused on binary relation extraction to the News domain, which is one of the goals of this thesis.

4 Methodology

This chapter presents the methodologies developed for this thesis. It starts by describing the selected Relation Extraction methods: Embedding Sub-sequences, Graph Convolutional Neural Network, and Transformers Neural Network. Then, the process of creating a Portuguese dataset in the News domain is explained. Finally, the quantitative and qualitative metrics are detailed in the last section.

4.1 Methods

From the literature review, a few techniques drew attention due to their simplicity and high performance. In this section, the selected approaches to the Relation Extraction task within the News context will be explained.

4.1.1 Embedding Sub-sequences

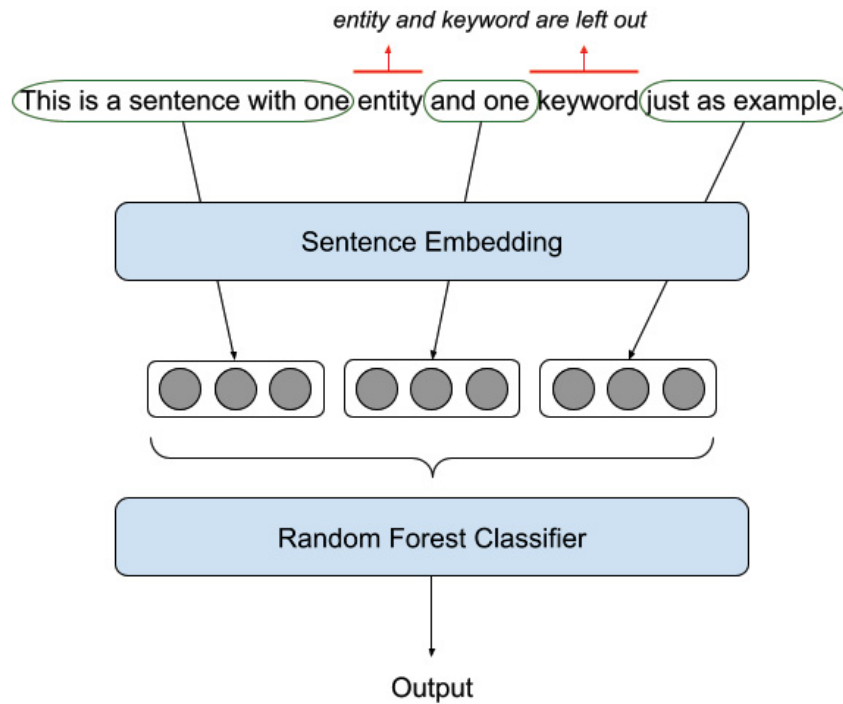
The Embedding Sub-sequences approach, illustrated on Figure 4.1, is inspired by Mooney & Bunescu (2005) proposed methodology. In their article, information about the sub-sequences around the entities (before, between, and after the entities) are extracted, such as the actual words and part-of-speeches. It follows the idea that the information about the two entities' relation must lie in their surroundings. Then, this information is passed to a kernel function, responsible for predicting relations.

To make use of a more recent and powerful tool, for this project the information extracted from the entities' surroundings comes from a pre-trained Portuguese embedding model, which is available in spaCy's library (HONNIBAL; MONTANI, 2017). This model is trained over thousands of Portuguese news and media text and uses the Continuous-Bag-Of-Words methodology, where it's able to take the context of the sentence into account. In other words, the entities' surroundings are converted into contextualized vectors using the embedding model, which can be concatenated into a single vector and used as input to a classification algorithm. Equation 4.1 expresses this step, where v_{concat} is the concatenated vector, and $v(s_{before})$, $v(s_{between})$ and $v(s_{after})$ are the embedding vectors of the sub-sequences before, between, and after the entities, respectively.

$$v_{concat} = (v(s_{before}), v(s_{between}), v(s_{after})) \quad (4.1)$$

Finally, to predict whether the entities in a sentence contain any relation, the simple but powerful Random Forest classifier was chosen, which was introduced by Ho (1995). This method consists of a large number of individual decision trees that operate

Figure 4.1 – Embedding Sub-sequences methodology.



Source: The author (2022).

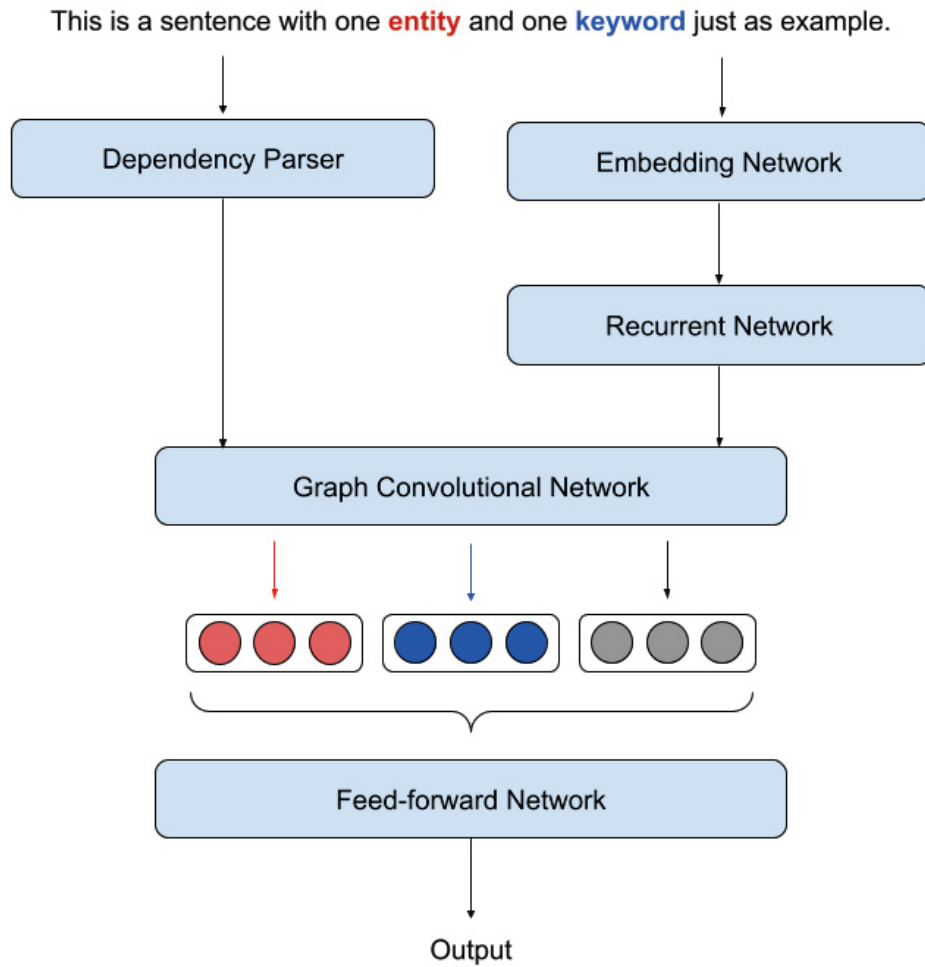
as an ensemble. Each individual tree in the random forest spits out a class prediction and the class with the most votes becomes the model's prediction.

4.1.2 Graph Convolutional Neural Network

Dependency parses of the input sentences have proven to be very effective in relation extraction because they capture long-range syntactic relations that are obscure from the surface alone. Based on Zhang, Qi & Manning (2018), a Graph Convolutional Neural Network (GCNN) was built for this project. This model encodes the dependency structure over the input sentence with efficient graph convolution operations, then extracts entity-centric representations to make robust relation predictions. Figure 4.2 illustrates this approach.

The first step of this architecture is to extract the dependency structure of the input sentence. For that, a fast and simple pre-trained dependency parse tool for the Portuguese language from spaCy was used. To be further used inside the neural network, the dependency structure must be represented as an adjacency matrix A , with shape $N \times N$, where N is the number of words in the sentence. If there is a connection between

Figure 4.2 – Graph Convolutional Neural Network methodology.



Source: The author (2022).

word i and word j , $A_{ij} = 1$, otherwise $A_{ij} = 0$.

With the same spaCy library used before, more information can be extracted and used to enrich the model: (i) the embedding word vector for each of the words in the sentence; (ii) the part-of-speech tags of the words; and (iii) the named entities. With all this information, an embedding neural network was built and trained along the whole architecture, providing contextualized information to the model. Moreover, the output of the embedding network was connected to a Recurrent Neural Network, more specifically a Bi-directional Long Short-term Memory, to insert even more contextualized information about word order and mitigate possible incorrect parse trees.

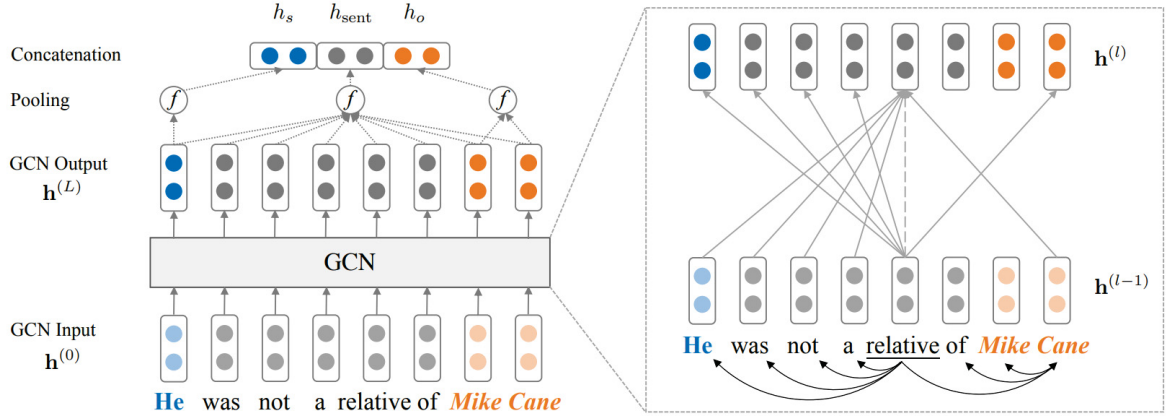
The Graph Convolutional Network takes the adjacency matrix, as well as the

output from the recurrent network to perform a series of convolutional operations. Each convolutional layer follows the Equation 4.2, where $h_i^{(l-1)}$ is the input vector and $h_i^{(l)}$ is the output vector of node i in the l -th layer, $W^{(l)}$ is a linear transformation, $b^{(l)}$ is a bias term, and σ is a nonlinear function.

$$h_i^{(l)} = \sigma \left(\sum_{j=1}^n A_{ij} W^{(l)} h_j^{(l-1)} + b^{(l)} \right) \quad (4.2)$$

By the end of the graph convolutional operations, hidden representations of each input token are obtained, which are directly influenced by their neighbors. With all the hidden representations, a max pooling function can map the output vectors to the complete sentence vectors. In the same way, the specific entities being evaluated in the Relation Extraction task can also be extracted as representation vectors, which is important since the information close to the entities in the dependency tree is often central to relation classification. Figure 4.3 illustrates this process.

Figure 4.3 – Graph Convolutions.



Source: Zhang, Qi & Manning (2018).

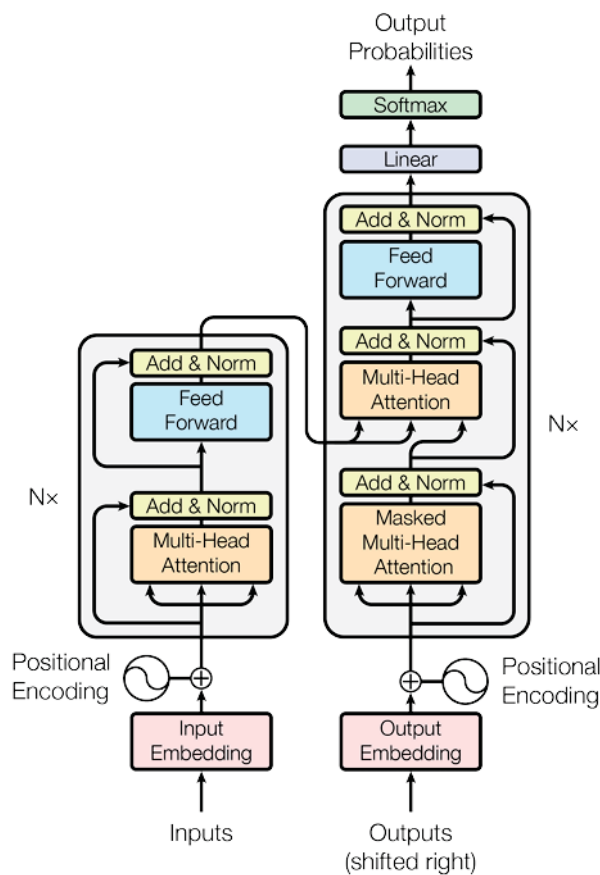
Finally, to obtain a probability distribution over the possible relations, the output vectors from the sentence and entity tokens are concatenated and fed to a feed-forward neural network, followed by a softmax operation.

4.1.3 Transformer Neural Network

The Transformer approach is based on Zhou & Chen (2021). As represented on Figure 4.4, Transformers are neural network models based solely on attention mechanisms

to draw global dependencies between input and output. They are the current state-of-the-art in sequence modeling and machine translation problems, beating (in terms of speed and accuracy) recurrent and convolutional neural networks. These previous architectures have some information bottleneck of the intermediary state caused by long sequence modeling, where oftentimes these models forget the first part of the sequence by the time it completes processing the whole input. The attention mechanism tries to overcome this problem by allowing the decoder model to access all the hidden states rather than a single vector to predict each output.

Figure 4.4 – Transformers architecture.

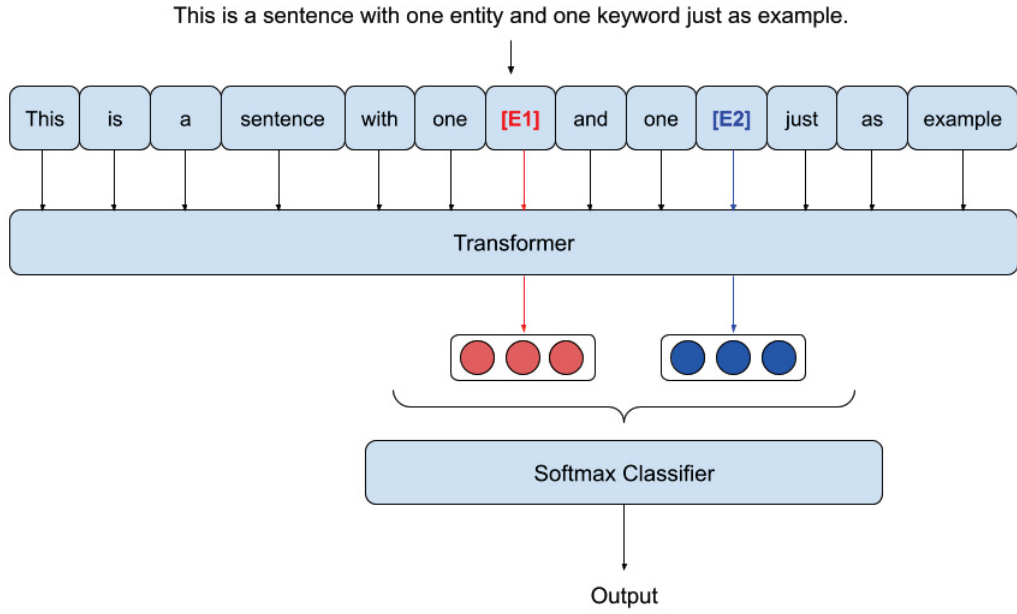


Source: Vaswani et al. (2017).

Figure 4.5 illustrates the proposed methodology of this project. The model starts by splitting the sentence into a list of tokens. This step makes use of a pre-trained Portuguese word piece model, which works by splitting words either into the full forms (e.g., one word becomes one token) or into word pieces — where one word can be broken

into multiple tokens. Although Transformers is one of the state-of-the-art models from sequence modeling, they are not pre-designed to handle Relation Extraction tasks. Soares et al. (2019) explored different methods of representing relations within the model. The methodology proposed here uses entity markers, replacing the entities in the sentence with generic tokens.

Figure 4.5 – Transformers methodology.



Source: The author (2022).

The tokens are fed to a pre-trained Portuguese Transformer model, called BERTimbau (SOUZA; NOGUEIRA; LOTUFO, 2020), which was trained on the Brazilian Web as Corpus, a large Portuguese corpus, for 1,000,000 steps, using whole-word masks. The hidden states from the last layer of the language model are then used as contextual embeddings of subtokens, which are sequences of dense vectors. These subtokens are concatenated into one vector, expressed by Equation 4.3, where v_{concat} is the concatenated vector, and $v(marker_1)$ and $v(marker_2)$ stands for the embedding vectors from the entity marker 1 and 2, respectively.

$$v_{concat} = (v(marker_1), v(marker_2)) \quad (4.3)$$

Finally, the concatenated vector is fed into a softmax classifier that returns a relation probability. During training, the Transformer, as well as the classifier are fine-tuned with the cross-entropy loss.

4.2 Dataset

The goal of this thesis is to evaluate the effectiveness of Relation Extraction techniques applied to the News domain, such as the process of looking for negative information associated with a name (individual or corporate) within news articles, legal prosecution, or similar content. Since no Portuguese benchmark was found in this context, this section will explain how a sentence-level binary relation extraction dataset was built, but because of Industrial Property, this benchmark will not be open-sourced.

4.2.1 News Sources

Figure 4.6 illustrates the whole process. The first step in building a dataset to extract relations from news is to find text that contains some kind of relations. One easy way of finding these texts is through web searches. For that, dozens of well-known media names, along with strategic keywords, were passed to the web search engines.

With the news pages from the web search, one challenge is to extract the correct content from these pages, since it contains much more than just the news text. For instance, it may contain headers, footers, other articles, images, advertising, and so on. Not only that, but the pages come from different providers, which have different web structures, making content extraction even more difficult.

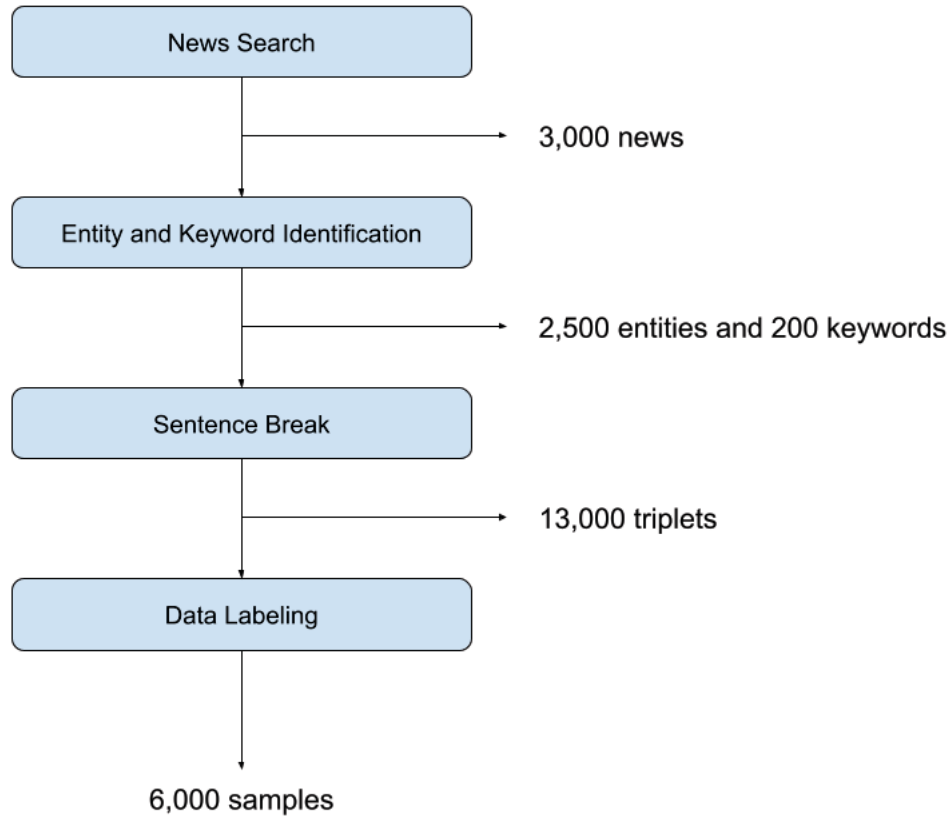
To overcome this difficulty, a complex web scrapper was built to collect the text from the web search results. By the end of this process, more than 3,000 unique piece of news were collected.

4.2.2 Preparation

For this project, relation extraction will be performed between entities and negative keywords. To identify the entities within the collected news articles, the spaCy library was selected due to its capability of Named Entity Recognition (NER) for the Portuguese language. Moreover, a simple regular expression technique was applied in every news article collected to find the negative keywords listed previously. By the end of this process, more than 2,500 unique entities and 200 distinct keywords were identified.

Although many facts are expressed by long context consisting of multiple sentences, this project only adopted relations inside the same sentence. The first reason is that a large source of news is available, where collecting samples wouldn't be a problem.

Figure 4.6 – Dataset methodology.



Source: The author (2022).

The second reason is due to the complexity of the system, where relations between words from different sentences usually need some kind of co-reference to link one another, which could impact performance. Finally, it was noticed that almost 80% of existing relations in the corpus are within the sentence level.

To split the news text into sentences, the spaCy library was also used due to its capability of identifying sentence breaks. For every sentence, each entity and negative keywords were combined. At this point, around 13,000 unique triplets (sentence, entity, keyword) were selected.

4.2.3 Data Labeling

Despite the success of existing RE methods, these approaches rely on a large number of human annotations, which can be expensive and time-consuming. To label the triples previously collected, a workforce of human labor was demanded, where each sentence was assigned to only one person and, on average, 200 samples could be annotated

in one hour.

The labeling methodology was defined as a classification problem with three possible labels: (i) the entity and keyword are negatively related; (ii) the entity and keyword are not negatively related; and (iii) the sentence should be discarded. This last label should be considered when:

- the entity is not a person, company, or public institution
- the sentence is not complete

During the labeling process, a few rounds of double-checking were performed, where one person re-annotated some samples from another person. By comparing the labels, 95% of the samples reached the same agreement, showing that the labelers understood the task in the same way, not requiring more than one person to label each sample, speeding up the process and making it less expensive.

The whole labeling process took around 30 days. In the end, 7,000 samples were discarded. The 6,000 remaining samples formed the final dataset. Since the labels were pretty much balanced, no further work was needed.

4.3 Evaluation Metrics

To evaluate the proposed methodologies, two approaches will be used: F1-score and Shapley Additive Explanations (SHAP). The former is a statistical measure that outputs a score. The latter is a black-box model explainability. Both will be explained in this section.

4.3.1 F1 score

In the automation of extracting relations in news, two metrics that should be optimized: (i) precision, guaranteeing that the news selected contains actual relation between entity and keyword; and (ii) recall, rejecting only the news that doesn't contain relations between entity and keyword. This information can be extracted by looking at the confusion matrix, which is a table that is used to visualize and summarize the performance of a classification algorithm. Figure 4.7 shows an example of a confusion matrix, where:

- True Positives (TP): the number of actual positive samples that were predicted to be positive.
- False Positives (FP): the number of actual negative samples that were predicted to be positive.

- False Negatives (FN): the number of actual positive samples that were predicted to be negative.
- True Negatives (TN): the number of actual negative samples that were predicted to be negative.

Figure 4.7 – Confusion matrix.

True Class	Positive	TP	FP
	Negative	FN	TN
		Positive	Negative
		Predicted Class	

Source: The author (2022).

Using the terms from the confusion matrix explained above, Equation 4.4 shows how the precision score is calculated, which is the ratio between the correct samples over the selected samples. And, Equation 4.5 shows how the recall score is calculated, which is the ratio between the correct samples over the possible correct samples.

$$precision = \frac{TP}{TP + FP} \quad (4.4)$$

$$recall = \frac{TP}{TP + FN} \quad (4.5)$$

To take both scores into account, the F1 metric was selected. Its formula is expressed in Equation 4.6, which is a harmonic mean between precision and recall metrics.

$$F1 = 2 \cdot \frac{precision \cdot recall}{precision + recall} \quad (4.6)$$

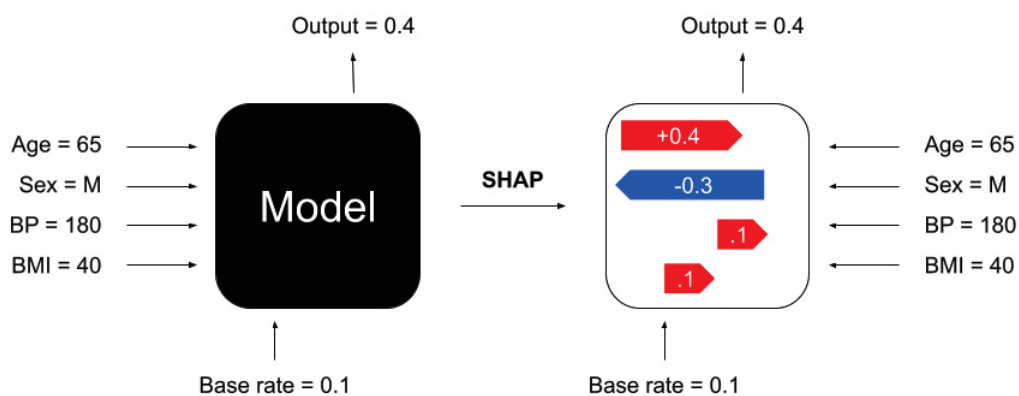
As seen in Chapter 3, F1-score is a common metric used to evaluate relation extraction problems. It makes sure that both inner metrics are high to have a high F1 score.

4.3.2 Shapley Additive Explanations

Understanding why a model makes a certain prediction can be as crucial as the prediction's accuracy in many applications. However, the highest accuracy for large modern datasets is often achieved by complex models that even experts struggle to interpret, such as ensemble or deep learning models, creating a tension between accuracy and interpretability. To address this problem, Lundberg & Lee (2017) presented SHAP (SHapley Additive exPlanations), which is a game theoretic approach to explain the output of any machine learning model by connecting optimal credit allocation with local explanations using the classic Shapley values from game theory and their related extensions.

Figure 4.8 illustrates a black box model that takes four values as input and outputs a prediction. After SHAP, the influence of every input on the final prediction is calculated, where the sum of the influences (plus the base rate) is equal to the output prediction.

Figure 4.8 – Shapley Additive Explanations.



Source: The author (2022).

The goal of SHAP is to explain the prediction of an instance x by computing the contribution of each feature to the prediction. The SHAP explanation method computes

Shapley values from coalitional game theory, where the feature values of a data instance act as players in the coalition. Shapley values tell us how to fairly distribute the “payout” (prediction) among the features. A player can be an individual feature value, e.g. for tabular data, or a group of feature values, for example, to explain an image, pixels can be grouped into superpixels and the prediction distributed among them.

One innovation that SHAP brings to the table is that the Shapley value explanation is represented as an additive feature attribution method, a linear model. Equation 4.7 expresses this method, where g is the explanation model, ϕ_0 is the base rate, z' is the coalition vector, M is the maximum coalition size, and ϕ_j is the feature attribution for a feature j (the Shapley values).

$$g(z') = \phi_0 + \sum_{j=1}^M \phi_j z'_j \quad (4.7)$$

Moreover, Python’s implementation of SHAP has an easy-to-use tool for text explanations. It is not specifically for relation extraction problems, but it provides various customization, enabling the adaptation for these kinds of problems.

4.4 Final Chapter Considerations

To prevent the overfit on specific words, and increase the domain adaptation through different news articles, the three Relation Extraction methodologies proposed in this chapter were modified to not take the actual entities into account. Even though each one is based on different papers, and have different strategies, all of them were able to ignore or replace the entities for generic tokens without disturbing the algorithm concept.

Another important consideration is that the development of the new Portuguese dataset makes it difficult to compare the current study with previous papers, since each language and benchmark has its own characteristics. However, some approximate comparison can be made when papers focus on similar tasks, which in this case consists on: two entities, a binary output, texts limited to the sentence-level and in the news domain.

5 Application

This chapter presents the results of applying the previously explained methods to the created dataset. To compare them, besides evaluating the F1 score and SHAP metrics, two other techniques were used across all applications: cross-validation and probability distribution plot.

Cross-validation is a resampling method that uses different portions of the data to train and test a model on different iterations. It is mainly used in settings where the goal is prediction, and one wants to estimate how accurately a predictive model will perform in practice, i.e., in unseen data. For these applications, since some methods can take longer to train, a 5-fold cross-validation was chosen, where each fold contains an 80/20 sample split without replacement.

A probability distribution plot is a visualization tool to help the understanding of how the predicted probabilities are distributed along each of the classes. It uses kernel density estimation (KDE) to estimate the probability density functions.

As mentioned in the previous chapter, SHAP will be used to better understand which input features drive the model towards a decision. For every method, a few positive and negative examples will be shown. Tokens [E1] and [E2] will show the boundaries of entity 1 and entity 2, respectively. The color will represent what class the word is driving the model to blue for negative class, and red for positive class. Finally, the color transparency will represent how much the word is driving the model to a decision.

Moreover, the training time (time to complete the training process) will also be recorded to be further compared among the methods. Every method was run on an environment provided by Kaggle, which was an Intel(R) Xeon(R) with 13 GB of RAM and 16 CPU. For the GCNN and Transformer method, GPU was also enabled, which had 16 GB of memory.

5.1 Embedding Sub-sequences

The Embedding Sub-sequences method implemented for this project uses a pre-trained model for word embedding and a Random Forest algorithm as the classification technique. The hyperparameters were set to the default values from the used libraries, which had the following configuration:

- Pre-trained word embedding: *pt_core_news_lg*
- Number of trees: 100

- Criterion: gini
- Maximum depth of the tree: unlimited
- Minimum samples to split a node: 2
- Minimum samples to be at a leaf: 1
- Minimum weight fraction: 0
- Maximum leaf nodes: unlimited
- Minimum impurity decrease: 0
- Bootstrap: true
- OOB score: false

Table 5.1 shows the scores and elapsed training times of each fold. Although the F1 score reached a maximum value of 74.82, it has some variation from its lowest score, which was 68.72, more than 6 percentage points. Both metrics, precision, and recall contribute to this variation, but mainly it comes from the recall score.

By the observed score variations, one can conclude that the model is a bit unstable, being easily biased by the samples during the training stage. One possible solution to this problem would be adding more samples to improve the generalization of the model.

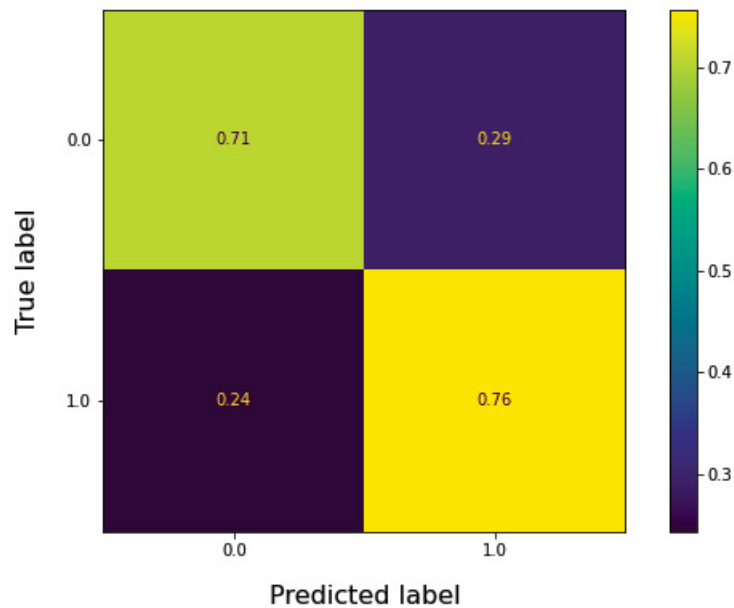
Table 5.1 – Embedding Sub-sequences Scores.

	Precision	Recall	F1	Training Time (min)
Fold 1	77.76	67.44	72.23	2:23
Fold 2	77.16	72.61	74.82	2:23
Fold 3	74.02	74.51	74.26	2:24
Fold 4	71.22	66.38	68.72	2:23
Fold 5	78.21	64.42	70.65	2:24
Mean	75.67	69.07	72.13	2:24
Std	2.98	4.29	2.53	0:03

Source: The author (2022).

Figure 5.1 shows the confusion matrix of the Embedding Sub-sequences model. Since the highest values are towards the True Positive and True Negative, the model was able to capture some patterns between the classes, even though there were a lot of False Positives and False Negatives as well.

Figure 5.1 – Embedding Sub-sequences Confusion Matrix.

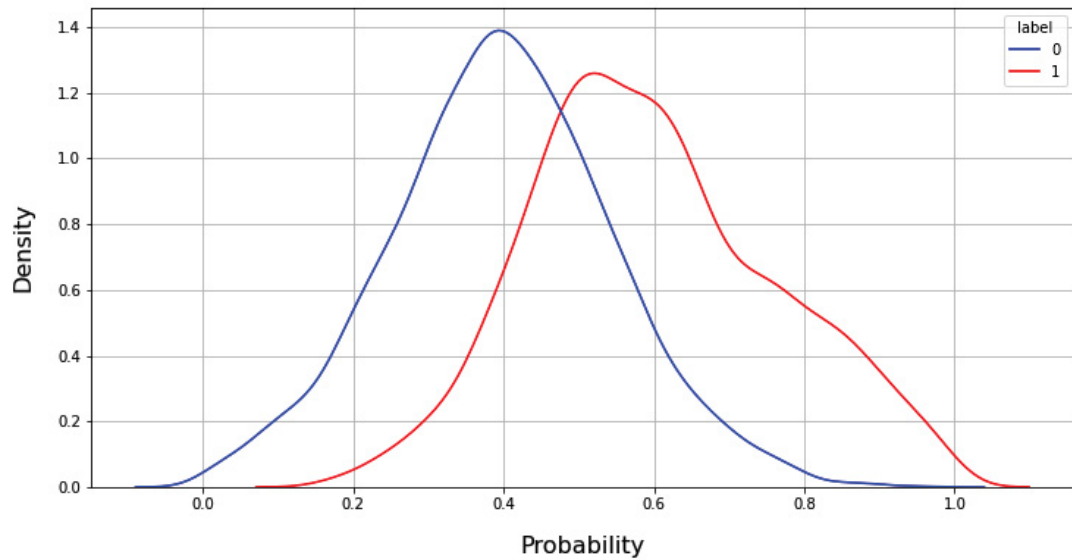


Source: The author (2022).

Figure 5.2 shows the Embedding Sub-sequences probability distribution plot. Again, it shows how this approach was not able to learn a decision boundary that could distinguish positive and negative samples with clarity. Although negative and positive samples have different distributions, it still has a lot of overlapping, resulting in misclassification.

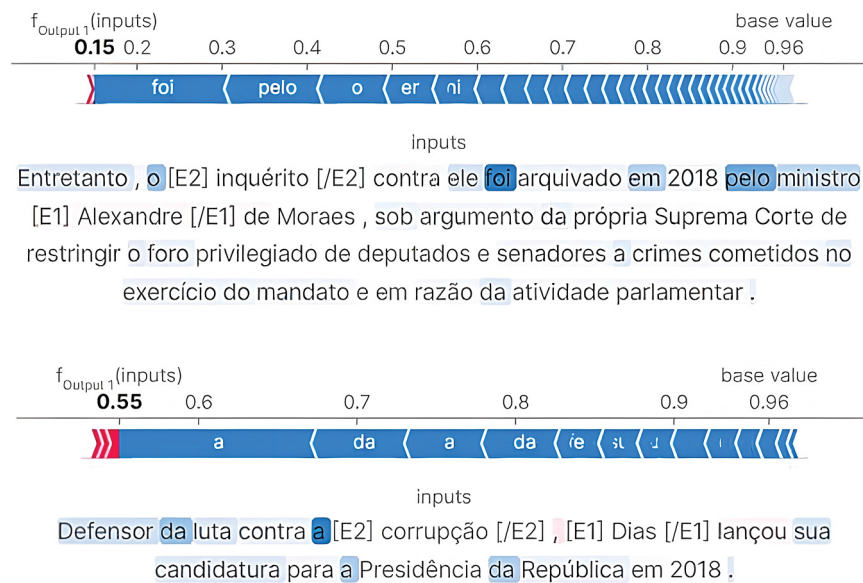
Figure 5.3 shows a couple of negative samples with the Shapley values by the Embedding Sub-sequences approach. In the first sentence, it's interesting to notice how the model could correctly focus on words close to the entities, leaving irrelevant and long distant context out from the final decision. However, in the second sentence, the model could not pick up enough negative words to correctly predict that the sample was negative.

Figure 5.2 – Embedding Sub-sequences Probability Plot.



Source: The author (2022).

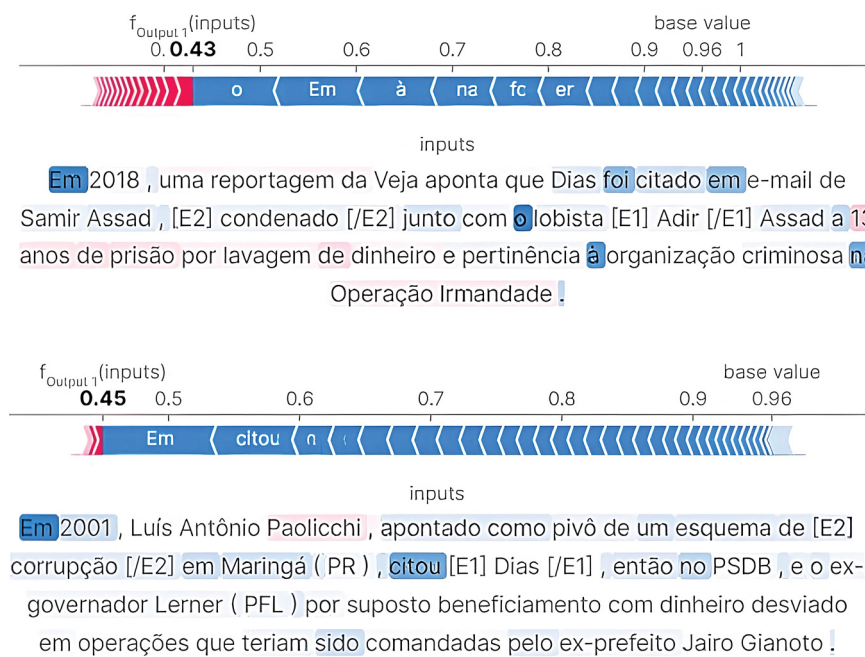
Figure 5.3 – Embedding Sub-sequences Negative Samples Shapley Values.



Source: The author (2022).

Figure 5.4 shows a couple of positive samples with the Shapley values by the Embedding Sub-sequences approach. In the first sentence, the model incorrectly gives too much attention to prepositions, driving the output towards the negative class. In the second sentence, the model considered too many words in the negative context, incorrectly predicting the output.

Figure 5.4 – Embedding Sub-sequences Positive Samples Shapley Values.



Source: The author (2022).

In both cases above, the base value is the same, 0.96. It means this model starts by assuming that a sentence is highly positive, and will only change its decision if enough negative information can be found within the sentence.

5.2 Graph Convolutional Neural Network

The Graph Convolutional Neural Network method implemented for this project uses a pre-trained model from spaCy to extract the dependency structure from text, as well as part-of-speech tags, followed by a combination of neural network layers. The hyperparameters were set based on the GCNN proposed by Zhang, Qi & Manning (2018), which had the following configuration:

- Pre-trained word embedding: *pt_core_news_lg*
- Batch size: 50
- Number of epochs: 100
- Learning rate: 0.3
- Learning rate decay: 0.98
- Gradient clipping: 5
- L2 regularization: 0
- L2 pooling: 0.003
- Part-of-speeches embedding size: 30
- Named entity embedding size: 30
- RNN hidden size: 200
- RNN number of layers: 1
- CNN hidden size: 200
- CNN number of layers: 2
- Classifier hidden size: 200
- Classifier number of layers: 4
- Dropout rate: 0.5

Table 5.2 shows the scores and elapsed training time of each fold. F1 score reached a maximum value of 85.08, especially due to its high recall score, which was 90.95. On the other hand, the precision score held the F1 score down, meaning that the model had some problems with False Positive predictions. By the observed score variations, one can conclude that the model reached similar values between folds, showing stable results and significant performances.

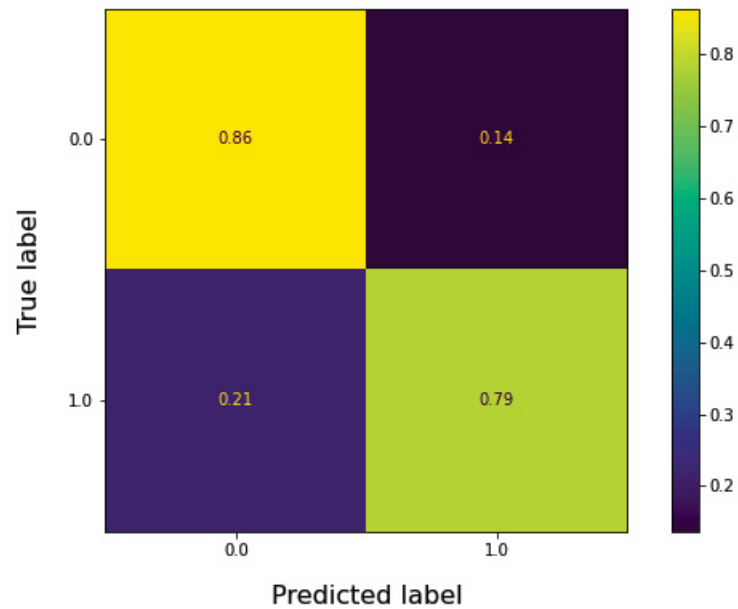
Figure 5.5 shows the confusion matrix of the GCNN model. Since the highest values are towards the True Positive and True Negative, the model was able to capture patterns between the classes. However, False Positive and False Negative values are still significant, which means there's still room for improvement.

Table 5.2 – Graph Convolutional Neural Network Scores.

	Precision	Recall	F1	Training Time (min)
Fold 1	77.06	89.66	82.88	8:30
Fold 2	80.59	86.62	83.50	9:01
Fold 3	77.29	88.98	82.72	9:08
Fold 4	77.93	85.74	81.65	9:01
Fold 5	79.92	90.95	85.08	8:51
Mean	78.55	88.39	83.16	8:56
Std	1.60	2.16	1.26	0:17

Source: The author (2022).

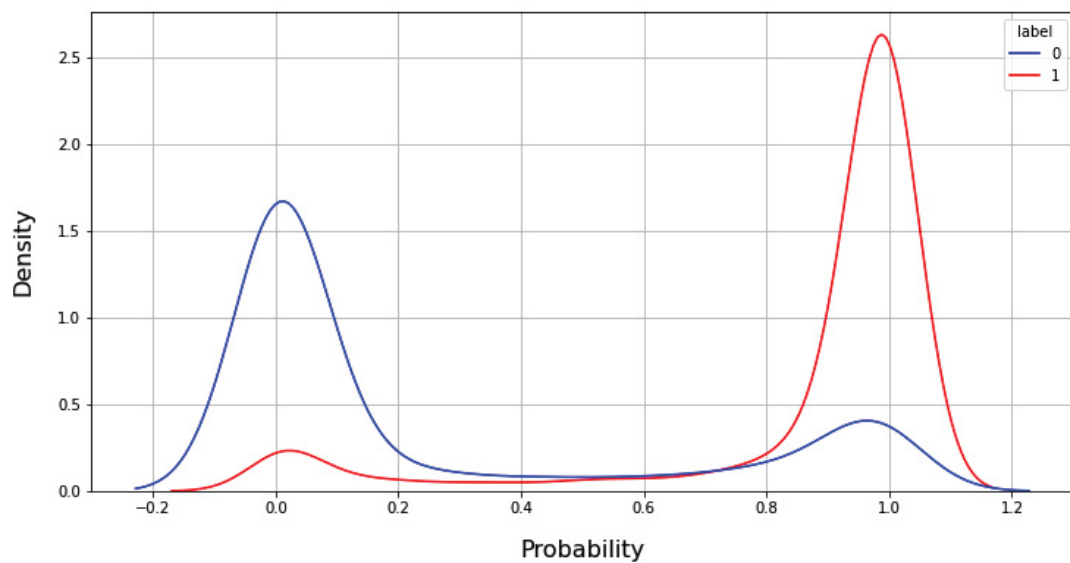
Figure 5.5 – Graph Convolutional Neural Network Confusion Matrix.



Source: The author (2022).

Figure 5.6 shows the GCNN probability distribution plot. It gives the understanding that this model has a lot of confidence when predicting the samples, whether it's positive or negative since the probability of the classes is either very high or very low. Nevertheless, the False Positives and False Negatives samples have high confidence as well, indicating that some learned patterns are misleading.

Figure 5.6 – Graph Convolutional Neural Network Probability Plot.

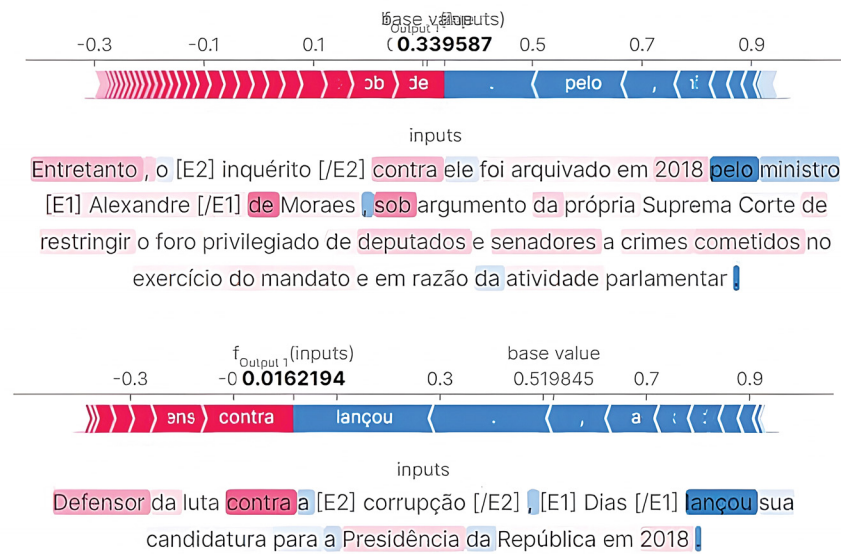


Source: The author (2022).

Figure 5.7 shows a couple of negative samples with the Shapley values by the GCNN approach. In the first sentence, the model could correctly predict the negative sentence especially due to the understanding that entity 1 was a Minister of Justice working on the case. In the second sentence, it is interesting how the model is placing high negative values on punctuation marks, which correctly separates two different contexts in the sentence.

Figure 5.8 shows a couple of negative samples with the Shapley values by the GCNN approach. In the first sentence, lots of words with negative context correctly drove the decision towards the positive class. However, in the second sentence, the words with negative context were not enough, letting the model incorrectly predict the output.

Figure 5.7 – Graph Convolutional Neural Network Negative Samples Shapley Values.



Source: The author (2022).

Figure 5.8 – Graph Convolutional Neural Network Positive Samples Shapley Values.



Source: The author (2022).

In both cases above, the base values are slightly closer to the negative class. It means this model starts the evaluation of the sentence with a small bias, but the decision will be given to the class with the most significant values.

5.3 Transformer Neural Network

The Transformer Neural Network method implemented for this project uses a pre-trained neural network, plus a final softmax classification layer. The hyperparameters were set based on the framework proposed by Zhou & Chen (2021), which had the following configuration:

- Pre-trained language model: Neuralmind’s *bert-large-portuguese-cased*
- Maximum sequence tokens: 512
- Warmup ratio: 0.1
- Optimization technique: Adam
- Optimization technique epsilon: 1e-6
- Gradient clipping: 1
- Gradient accumulation steps: 2
- Batch size: 32
- Dropout rate: 0.1
- Learning rate: 3e-5
- Number of epochs: 5

Table 5.3 shows the scores and elapsed training times of each fold. F1 score reached a maximum value of 93.12. The recall score reached a maximum value of 94.75. Not far behind, precision’s highest score was 92.50. All of these values indicate that the model was able to capture most of the contextualized patterns, resulting in few False Positives and few False Negatives.

Moreover, by the observed score variations, one can conclude that the model reached similar values between folds, showing stable results. Not only that, but all folds achieved significant high values.

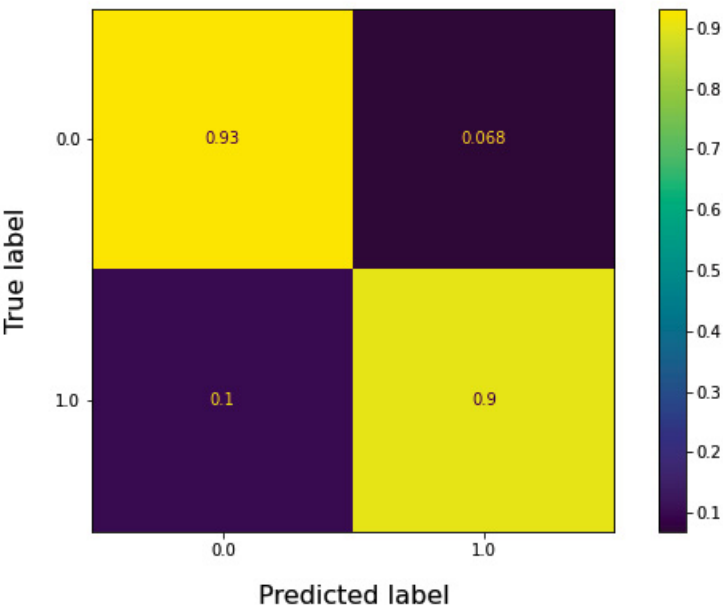
Table 5.3 – Transformer Neural Network Scores.

	Precision	Recall	F1	Training Time (min)
Fold 1	89.64	94.75	92.12	24:17
Fold 2	92.50	92.36	92.43	22:04
Fold 3	88.17	94.41	91.18	24:23
Fold 4	87.82	91.85	89.79	24:49
Fold 5	91.80	94.48	93.12	22:59
Mean	89.98	93.57	91.72	23:42
Std	2.10	1.36	1.29	1:20

Source: The author (2022).

From the confusion matrix in Figure 5.9, one can conclude that this approach is very good in predicting either positive relations or negative relations since in both cases it reached a score of at least 0.9.

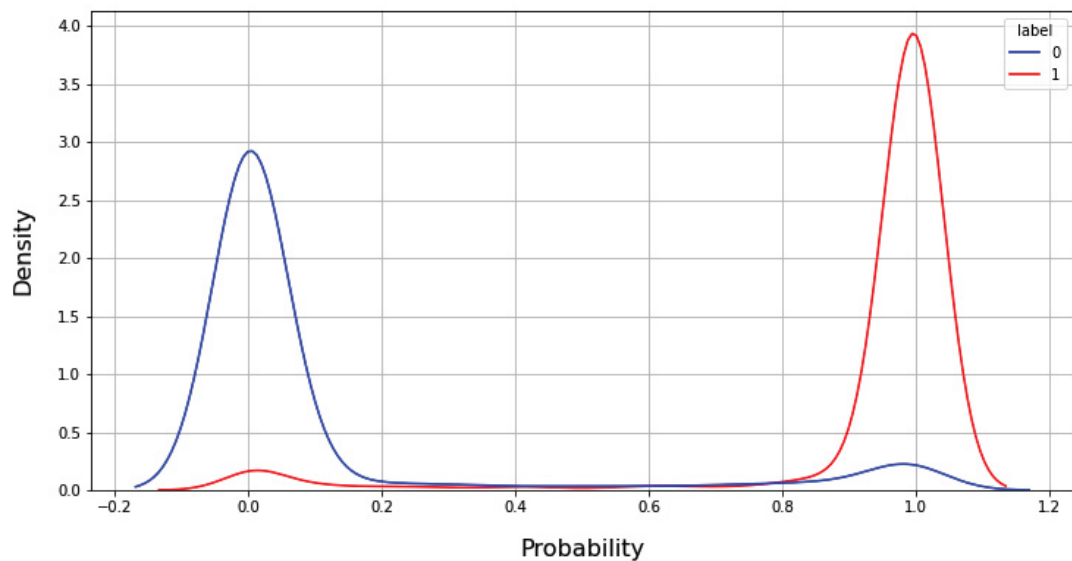
Figure 5.9 – Transformer Confusion Matrix.



Source: The author (2022).

Figure 5.10 shows the probability distribution plot. With this, it's possible to conclude that the Transformer model has a lot of confidence in predicting the samples since the probability of the classes is either very high or very low. Furthermore, the density of the misclassifications is extremely low, compared to the density of the correctly classified.

Figure 5.10 – Transformer Probability Plot.

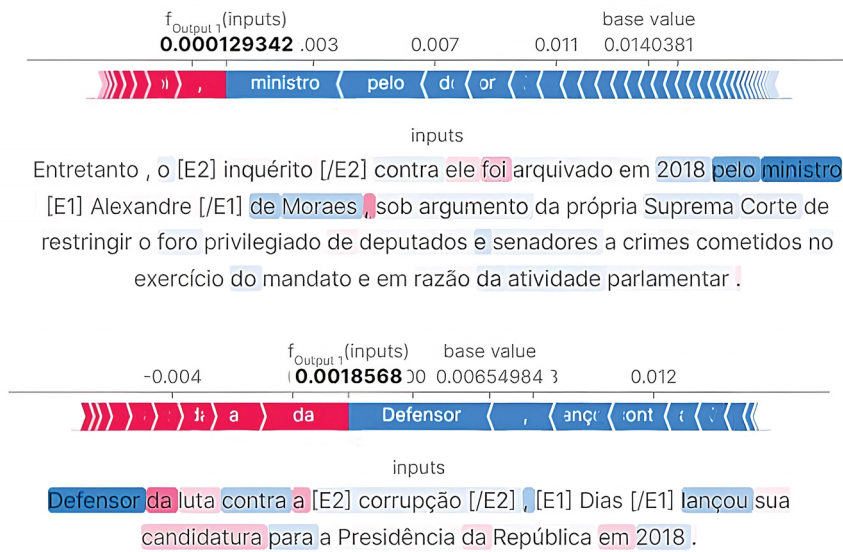


Source: The author (2022).

Figure 5.11 shows a couple of negative samples with the Shapley values by the Transformer approach. In the first sentence, the model could correctly predict the negative sentence especially due to the understanding that entity 1 was a Minister of Justice working on the case. In the second sentence, the model could correctly understand that entity 1 is positively related to entity 2, even though entity 2 is a bad media.

Figure 5.12 shows a couple of negative samples with the Shapley values by the Transformer approach. In the first sentence, the negative words between the entities were not enough, resulting in an incorrect prediction from the model. However, in the second sentence, the few words around the entities were more than enough for the model to output the correct prediction.

Figure 5.11 – Transformer Negative Samples Shapley Values.



Source: The author (2022).

Figure 5.12 – Transformer Positive Samples Shapley Values.



Source: The author (2022).

In both cases above, the base values are very small. It means this model starts by assuming that a sentence is highly negative, and will only change its decision if enough positive information can be found within the sentence.

5.4 Final Chapter Considerations

In the previous sections, the results from the application of three different methodologies in the Portuguese News context were presented separately. In this section, the results will be discussed together, along with the information found in papers during the literature review process.

Based on the mean F1 score obtained from the methods during the cross-validation process, the Transformer Neural Network achieved the best result, reaching an incredible 91.72 score. The Graph Convolutional Neural Network also achieved a significant score of 83.16, followed by the modest Embedding Sub-sequences score of 72.13.

Even though the benchmarks used to evaluate the methodologies are different, the Transformer approach applied in this thesis reached interesting results when compared to the literature scores. A few choices may be responsible for this accomplishment: specific target domain, sentence-level granularity, and a binary output category, along with the powerful transfer-learning from Transformers.

On the other hand, from the elapsed training time, one can conclude that the Embedding Sub-sequences approach is the simplest algorithm, being four times faster than the GCNN, and 10 times faster than the Transformer Neural Network. These training times were achieved using GPU on all approaches but the Embedding Sub-sequences. Moreover, depending on the final application, simpler and faster methods may be a better choice, but for this thesis, longer training and prediction time were not an issue.

The probability plots shown for every approach highlight the effectiveness of the Transformer method, where the density curve of the classes is far apart from each other. The same happens in the GCNN plot but with a smaller magnitude. In contrast, the Embedding Sub-sequences probability plot shows how this model was unable to correctly separate the classes, resulting in a lot of misclassified samples.

Finally, the SHAP visualizations confirm the advantages of using the Transformer approach. This method seems to understand the context in the sentence, giving more importance to specific words near the entities. The GCNN method also highlights important words near the entities, but it doesn't give as much importance as the Transformer approach. Meanwhile, the Embedding Sub-sequences gives too much importance to words all over the sentence, not being able to capture the context of the sentence.

6 Final Considerations

Screening for negative news means identifying and monitoring a wide range of conventional printed and televised media content, along with online content like blogs and social media feeds. This should be an important part of any financial institution’s customer due diligence process to discover its client’s risk profile and protect itself from reputational damage.

Since this process traditionally involves manual checks and human interpretation, the sheer amount of information firms has to deal with during screening makes it labor-intensive and time-consuming. To manage the information collected, best practice guidance suggests using automation to complement the manual elements of the review process.

For this reason, this thesis started by understanding how Relation Extraction is commonly used. Different types of RE settings were discussed, including target domain, textual granularity, number of arguments, and semantic categories. Moreover, several approaches that have been developed over the year were explored, which can involve different types of learning, computational components, structures, and input features.

With the literature review, promising Relation Extraction solutions were found, from early kernel methods using statistical features, to recent neural networks that use huge pre-trained language models. Along with the state-of-the-art methodologies, the F1 score was the statistical metric that is widely used on RE problems. Moreover, a few benchmarks are common across papers, such as ACE, SemEval Task 8, and TACRED.

Based on the literature review, three different solutions were proposed to solve RE problems: (i) an Embedding Sub-sequences, that uses the embedding vectors from around the entities as input to a Random Forest classifier; (ii) a Graph Convolutional Neural Network, that uses embedding vectors along with dependency parse trees as input to convolutional layers; and (iii) a Transformer Neural Network, which uses a pre-trained neural network with attention mechanisms.

Since no specific Portuguese benchmark was found for the News domain, text from the news was manually collected, processed, and labeled. The Relation Extraction settings of the problem were limited to the sentence level, two arguments, and only two possible output categories (related or not related).

The three proposed approaches were then applied to the benchmark created on a 5-fold cross-validation, where the Transformer methodology showed results significantly better than the other approaches. For instance, the Transformer F1-score reached 91.72, which was 10% higher than the GCNN result, and 27% higher than the Embedding Sub-sequences result.

Moreover, SHAP was used to better understand the influence that each input feature (words) has on the predicted value. Once again, the Transformer Neural Network overcame the other methodologies by giving high attention to important words around the entities, while the other methodologies gave great incorrect attention to words all over the sentence.

Based on state-of-the-art results from Relation Extraction solutions, the approaches presented in this thesis reached interesting results, especially the Transformer Neural Network and the Graph Convolutional Neural Network. With this, the task of negative news search within the Portuguese language could be automated using Relation Extraction algorithms.

6.1 Future Works

For future works, to improve the results obtained in this thesis, a comparison of the hyperparameters change would be interesting, since in this project the values were fixed based on previous papers. Moreover, to better evaluate the performance of the methodologies presented here, one could also apply them to other Portuguese benchmarks, regardless of the corpus domain.

To explore novel methodologies, would be interesting to combine the Transformer Neural Network with the context extracted by Graph Convolutions. There are several ways to achieve that, one of them being the concatenation of the Transformer final layer output with the GCNN final layer output, feeding them to a final classification algorithm.

Finally, seeing how transfer-learning played an important part in the success of the Transformer approach, and since most of the advantages come in the English domain first, one could evaluate the results obtained by translating the Portuguese documents to English before feeding them to a trained model.

REFERENCES

- ABACHA, A. B.; ZWEIGENBAUM, P. Automatic extraction of semantic relations between medical entities: a rule based approach. *Journal of biomedical semantics*, BioMed Central, v. 2, n. 5, p. 1–11, 2011.
- ABREU, S. C. de; VIEIRA, R. Relp: Portuguese open relation extraction. *KO KNOWLEDGE ORGANIZATION*, Nomos Verlagsgesellschaft mbH & Co. KG, v. 44, n. 3, p. 163–177, 2017.
- AGGARWAL, N.; WAREHAM, S.; LEHMANN, R. Applications of machine learning in the identification, measurement and mitigation of money laundering. *Journal of Financial Compliance*, Henry Stewart Publications, v. 4, n. 2, p. 140–166, 2020.
- BIRD, S.; KLEIN, E.; LOPER, E. *Natural language processing with Python: analyzing text with the natural language toolkit*. [S.l.: " O'Reilly Media, Inc.", 2009.
- BLACKBURN, J. L.; DOMIN, T. J. *Protective Relaying: Principles and Applications*. 3th. ed. [S.l.: CRC Press, 2006. 638 p. ISBN 1574447165.
- BUNESCU, R. C.; MOONEY, R. J. A shortest path dependency kernel for relation extraction. In: *Proceedings of the conference on human language technology and empirical methods in natural language processing*. [S.l.: s.n.], 2005. p. 724–731.
- CHRISTOPOULOU, E. *Textual Relation Extraction with Edge-oriented Graph Neural Models*. Tese (Doutorado) — University of Manchester, 2020.
- FERNANDES, E. R.; MILIDIÚ, R. L.; RENTERÍA, R. P. Relhunter: a machine learning method for relation extraction from text. *Journal of the Brazilian Computer Society*, Springer, v. 16, n. 3, p. 191–199, 2010.
- FERREIRA, J.; OLIVEIRA, H. G.; RODRIGUES, R. Nlpyport: Named entity recognition with crf and rule-based relation extraction. In: *IberLEF@ SEPLN*. [S.l.: s.n.], 2019. p. 468–477.
- FUNDEL, K.; KÜFFNER, R.; ZIMMER, R. Rellex—relation extraction using dependency parse trees. *Bioinformatics*, Oxford University Press, v. 23, n. 3, p. 365–371, 2007.
- GARCIA, M.; GAMALLO, P. Evaluating various linguistic features on semantic relation extraction. In: *Proceedings of the International Conference Recent Advances in Natural Language Processing 2011*. [S.l.: s.n.], 2011. p. 721–726.
- GUO, Z.; ZHANG, Y.; LU, W. Attention guided graph convolutional networks for relation extraction. *arXiv preprint arXiv:1906.07510*, 2019.
- GUSENBAUER, M.; HADDAWAY, N. R. Which academic search systems are suitable for systematic reviews or meta-analyses? evaluating retrieval qualities of google scholar, pubmed, and 26 other resources. *Research synthesis methods*, Wiley Online Library, v. 11, n. 2, p. 181–217, 2020.

- HAN, J. et al. Nextgen aml: Distributed deep learning based language technologies to augment anti money laundering investigation. In: *Proceedings of ACL 2018, System Demonstrations*. [S.l.: s.n.], 2018. p. 37–42.
- HO, T. K. Random decision forests. In: IEEE. *Proceedings of 3rd international conference on document analysis and recognition*. [S.l.], 1995. v. 1, p. 278–282.
- HONNIBAL, M.; MONTANI, I. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear. 2017.
- KHANDPUR, R. P. et al. Adverse media mining for kyc and esg compliance. *arXiv preprint arXiv:2110.11542*, 2021.
- LEVY, O. et al. Zero-shot relation extraction via reading comprehension. *arXiv preprint arXiv:1706.04115*, 2017.
- LIU, S. et al. Drug-drug interaction extraction via convolutional neural networks. *Computational and mathematical methods in medicine*, Hindawi, v. 2016, 2016.
- LUNDBERG, S. M.; LEE, S.-I. A unified approach to interpreting model predictions. In: GUYON, I. et al. (Ed.). *Advances in Neural Information Processing Systems 30*. Curran Associates, Inc., 2017. p. 4765–4774. Disponível em: <http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf>.
- MIKOLOV, T. et al. Distributed representations of words and phrases and their compositionality. In: *Advances in neural information processing systems*. [S.l.: s.n.], 2013. p. 3111–3119.
- MOONEY, R.; BUNESCU, R. Subsequence kernels for relation extraction. *Advances in neural information processing systems*, v. 18, 2005.
- NGUYEN, T. H.; GRISHMAN, R. Relation extraction: Perspective from convolutional neural networks. In: *Proceedings of the 1st workshop on vector space modeling for natural language processing*. [S.l.: s.n.], 2015. p. 39–48.
- PENNINGTON, J.; SOCHER, R.; MANNING, C. D. Glove: Global vectors for word representation. In: *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. [S.l.: s.n.], 2014. p. 1532–1543.
- PLANK, B.; MOSCHITTI, A. Embedding semantic similarity in tree kernels for domain adaptation of relation extraction. In: *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. [S.l.: s.n.], 2013. p. 1498–1507.
- REYES, D. D. L. et al. Entity relation extraction from news articles in portuguese for competitive intelligence based on bert. Springer, 2021.
- SANDHYA, S. Using word embeddings to explore the language of depression on twitter. 2019.
- SHEN, Y.; HUANG, X.-J. Attention-based convolutional neural network for semantic relation extraction. In: *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*. [S.l.: s.n.], 2016. p. 2526–2536.

- SOARES, L. B. et al. Matching the blanks: Distributional similarity for relation learning. *arXiv preprint arXiv:1906.03158*, 2019.
- SOUZA, F.; NOGUEIRA, R.; LOTUFO, R. BERTimbau: pretrained BERT models for Brazilian Portuguese. In: *9th Brazilian Conference on Intelligent Systems, BRACIS, Rio Grande do Sul, Brazil, October 20-23 (to appear)*. [S.l.: s.n.], 2020.
- VASWANI, A. et al. Attention is all you need. *Advances in neural information processing systems*, v. 30, 2017.
- VERGA, P.; STRUBELL, E.; MCCALLUM, A. Simultaneously self-attending to all mentions for full-abstract biological relation extraction. *arXiv preprint arXiv:1802.10569*, 2018.
- WENG, L. *Learning Word Embedding*. 2017. Disponível em: <https://lilianweng.github.io/posts/2017-10-15-word-embedding>.
- ZELENKO, D.; AONE, C.; RICHARDELLA, A. Kernel methods for relation extraction. *Journal of machine learning research*, v. 3, n. Feb, p. 1083–1106, 2003.
- ZHANG, A. et al. *Dive into Deep Learning*. [S.l.: s.n.], 2020. <https://d2l.ai>.
- ZHANG, Y.; QI, P.; MANNING, C. D. Graph convolution over pruned dependency trees improves relation extraction. *arXiv preprint arXiv:1809.10185*, 2018.
- ZHOU, G. et al. Exploring various knowledge in relation extraction. In: *Proceedings of the 43rd annual meeting of the association for computational linguistics (acl'05)*. [S.l.: s.n.], 2005. p. 427–434.
- ZHOU, W.; CHEN, M. An improved baseline for sentence-level relation extraction. *arXiv preprint arXiv:2102.01373*, 2021.