Pontifícia Universidade Católica
do Rio de Janeiro

Rafael dos Reis Silva

Direct and Indirect Quotation Extraction for Portuguese

Dissertação de Mestrado

Dissertation presented to the Programa de Pós-Graduação em Informática of PUC-Rio in partial fulfillment of the requirements for the degree of Mestre em Informática

Advisor : Prof. Ruy Luiz Milidiú
Co–Advisor: Prof. Maria Cláudia de Freitas

Rio de Janeiro
February 2017

PUC-Rio - Certificação Digital Nº 1421603/CA

Pontifícia Universidade Católica
do Rio de Janeiro

Rafael dos Reis Silva

Direct and Indirect Quotation Extraction for Portuguese

Dissertation presented to the Programa de Pós-Graduação em Informática of PUC-Rio in partial fulfillment of the requirements for the degree of Mestre em Informática. Approved by the undersigned Examination Committee.

Prof. Ruy Luiz Milidiú
Advisor
Departamento de Informática – PUC-Rio

Profa. Maria Cláudia de Freitas
Co–Advisor
Departamento de Letras – PUC-Rio

Prof. Hélio Cortês Vieira Lopes
Departamento de Informática – PUC-Rio

Prof. Leonardo Guerreiro Azevedo
IBM Research Brasil

Prof. Márcio da Silveira Carvalho
Coordinator of the Centro Técnico Científico – PUC-Rio

Rio de Janeiro, February 8th, 2017

PUC-Rio - Certificação Digital Nº 1421603/CA

All rights reserved.

Rafael dos Reis Silva

Graduated in 2008 from the Universidade Federal do Rio de Janeiro (UFRJ) in Computer Science. Joined the LEARN lab at the Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio) in 2015, focusing his research on Machine Learning and Natural Language Processing.

PUC-Rio - Certificação Digital Nº 1421603/CA

Bibliographic data

dos Reis Silva, Rafael

Direct and Indirect Quotation Extraction for Portuguese / Rafael dos Reis Silva; advisor: Ruy Luiz Milidiú; co–advisor: Maria Cláudia de Freitas. – Rio de Janeiro: PUC-Rio, Departamento de Informática, 2017.

v., 59 f: il. ; 29,7 cm

Dissertação (mestrado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Informática.

Inclui bibliografia

1. Informática – Teses. 2. Aprendizado de Máquina Supervisionado – Teses. 3. Processamento de Linguagem Natural. 4. Extração de Informação. 5. Extração de Citação. 6. Perceptron Estruturado. 7. Agendamento de Tarefas Ponderado. I. Milidiú, Ruy Luiz. II. de Freitas, Maria Cláudia. III. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. IV. Título.

CDD: 004

Acknowledgements

First and foremost, to my advisor, Ruy, for giving me the opportunity to work with him on this project, for his faith in me, and for his valuable mentoring during this period; to my co-advisor, *Claudinha*, for her creativity, passion and dedication to this research.

To my family, in particular, my parents and my sister, for their love, affection and understanding during this period of my absence.

To my *rolé* mate, Gustavo, for his genuine companionship, affection and for being so full of fun.

To all my friends at PUC, in particular, Sônia, for sharing this not-always-easy journey and all-important coffee breaks (and laughter), and Luis Felipe, for his friendship and productive machine learning discussions.

To my friend and English teacher, Rami, for her always-good mood, wise words and for helping me revise parts of this work.

To my colleagues at LEARN, in particular to Yanely, for her patience and teachings that saved me months of investigation.

To PUC-Rio and CNPq for their support.

To all of you, my sincere thanks!

PUC-Rio - Certificação Digital Nº 1421603/CA

Abstract

dos Reis Silva, Rafael; Milidiú, Ruy Luiz (Advisor); de Freitas, Maria Cláudia (Co-Advisor). **Direct and Indirect Quotation Extraction for Portuguese**. Rio de Janeiro, 2017. 59p. Dissertação de Mestrado – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Quotation Extraction consists of identifying quotations from a text and associating them to their authors. In this work, we present a Direct and Indirect Quotation Extraction System for Portuguese. Quotation Extraction has been previously approached using different techniques and for several languages. Our proposal differs from previous work, because we build a Machine Learning model that, besides recognizing direct quotations, it also recognizes indirect ones in Portuguese. Indirect quotations are hard to be identified in a text, due to the lack of explicit delimitation. Nevertheless, they happen more often then the delimited ones and, for this reason, have an huge importance on information extraction. Due to the fact that we use a Machine Learning model based, we can easily adapt it to other languages, needing only a list of verbs of speech for a given language. Few were the previously proposed systems that tackled the task of indirect quotations and neither of them for Portuguese using a Machine Learning approach. We build a Quotation Extractor using a model for the *Structured Perceptron* algorithm. In order to train and evaluate the system, we build QuoTrees 1.0 corpus. We annotate it to tackle the indirect quotation problem. The Structured Perceptron based on weight interval scheduling obtains an $F1$ score of 66% for QuoTrees 1.0 corpus.

PUC-Rio - Certificação Digital Nº 1421603/CA

Keywords

Machine Learning; Natural Language Processing; Information Extraction; Quotation Extraction; Structured Perceptron; Weighted Interval Scheduling;

Resumo

dos Reis Silva, Rafael; Milidiú, Ruy Luiz; de Freitas, Maria Cláudia. **Extração de Citações Diretas e Indiretas para o Português**. Rio de Janeiro, 2017. 59p. Dissertação de Mestrado – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

A Extração de Citações consiste na identificação de citações de um texto e na associação destas com seus autores. Neste trabalho, apresentamos um Extrator de Citações Diretas e Indiretas para o Português. A tarefa de Extração de Citações já foi abordada usando diversas técnicas em diversos idiomas. Nossa proposta difere das anteriores, pois construímos um modelo de Aprendizado de Máquina que, além de indetificar citações diretas, também identifica as citações indiretas. Citações indiretas são difíceis de serem identificadas num texto por não conter delimitações explícitas. Porém, são mais frequentes do que as delimitadas e, por essa razão, possuem grande importância na extração de informação. Por utilizarmos um modelo baseado em Aprendizado de Máquina, podemos facilmente adaptá-lo para outras línguas, bastando apenas uma lista de verbos do dizer num dado idioma. Poucos foram os sistemas propostos anteriormente que atacaram o problema das citações indiretas e nenhum deles para o Português usando Aprendizado de Máquina. Nós construímos um Extrator de Citações usando um modelo para o algoritmo do *Perceptron Estruturado*. Com o objetivo de treinar e avaliar o sistema, construímos o corpus QuoTrees 1.0. Nós anotamos este corpus a fim de atacar o problema das citações indiretas. O Perceptron Estruturado baseado no agendamento de tarefas ponderado tem desempenho $F1$ igual a 66% para o corpus QuoTrees 1.0.

PUC-Rio - Certificação Digital Nº 1421603/CA

Palavras-chave

Processamento de Linguagem Natural; Extração de Informação; Extração de Citação; Perceptron Estruturado; Agendamento de Tarefas Ponderado;

Table of contents

PUC-Rio - Certificação Digital Nº 1421603/CA

PUC-Rio - Certificação Digital Nº 1421603/CA

List of figures

PUC-Rio - Certificação Digital Nº 1421603/CA

List of tables

PUC-Rio - Certificação Digital Nº 1421603/CA

PUC-Rio - Certificação Digital Nº 1421603/CA

What Paul says about Peter tells us more about Paul than about Peter.

Baruch Spinoza.

1
Introduction

In the last decade, as a result of the Internet's growth, human kind has been producing a huge amount of data. Social networks, Big Data, the Internet of Things, to name just a few, are keywords which confirm that we are living at the peak of the Information Age. One kind of data that has a critical role in this scenario is textual data: besides the Web itself, emails and text messages provide an infinite source, in many different languages.

Researchers are trying to cope with this vast amount of digital information. The Artificial Intelligence field has been making great strides in processing this data to deliver meaningful knowledge in a short time. For instance, as the result of the involvement of the sub-field of Natural Language Processing (NLP), today we can get many sorts of information just by asking our mobiles phones out loud. This information can be as complex and sophisticated as our ability to process the textual ocean available on the Internet.

Since the 1980s, NLP has been successfully employing probabilistic models to discover common patterns that occur in language use [1]. In contrast to the rationalist approach, these new models proved to be adaptive to different writing styles, producing more reliable results. Later on, these models came to be known as Machine Learning (ML).

The ML paradigm uses general learning algorithms in order to analyze large volumes of data. Such algorithms learn patterns in data and apply them to new inputs.

In NLP, ML techniques are language independent, that is, we are able to apply to Portuguese a model originally prepared for English, with just a few changes.

Many classical ML algorithms are binary classifiers. These classifiers can be combined in compound systems that are able to predict complex structures. The naive approach of treating each structure as a separate class is often intractable, since it leads to a multiclass problem with a very large number of classes [2]. In the 2000s, however, some ML methods were proposed to solve *directly* structured problems, with great success [3–5]. They are called *structured learning* methods.

In order to facilitate information retrieval, we propose a Quotation

PUC-Rio - Certificação Digital Nº 1421603/CA

Extraction System for Portuguese, which consists of identifying quotations from a text and associating them with their authors. Our system handles direct and indirect quotations in Portuguese.

The proposed system is very useful in several situations. A voter may want to see what her candidate is saying in the media. An Almodóvar fan may want to see what he says about his last motion picture. We may test whether a newspaper article is unbiased by checking who has a say in it. There are many other situations our system may be applied to.

Quotation Extraction has been previously approached using different techniques and for several languages. Most of these methods are based on lexical or syntactic patterns. Just recently, some studies have applied supervised ML to this task [6–8].

Our proposal is the first work to address direct and indirect quotations in Portuguese using a Machine Learning approach. We extends [6], using the *Structured Perceptron* algorithm [4].

The Structured Perceptron is used to predict complex and interdependent outputs like sequences, trees and even more general graphs. The predictor is based on an optimization problem whose objective function is linear in the input-output feature vector. In the literature, algorithms of this kind are called structured learning methods.

Since we employ supervised ML algorithms, we need an annotated corpus to train and evaluate the system. In order to address the direct and indirect quotation problem, we create QuoTrees 1.0, the first corpus in Portuguese annotated with direct and indirect quotations features.

In our work, we create a ML model to tackle the Quotation Extraction task. This model is a Structured Perceptron, based on the weighted interval scheduling problem (SP-WIS). In this model, we find a maximum-weight interval subset of non-overlapping tasks, where each task represents a combination of quotation and author candidates.

In Table 1.1, we present the quality of our two models, assessed in the test set. SP-WIS obtains a new state-of-the-art F-score of 66% for QuoTrees 1.0, tackling direct and indirect quotation, in addition to *missing subject* quotes.

	Direct			Indirect			All		
Model	*P*	*R*	*F1*	*P*	*R*	*F*	*P*	*R*	*F*
with miss. subj.	.70	.67	.68	.88	.87	.87	.54	.79	.66
without miss. subj.	.80	.77	.78	.91	.91	.91	.72	.86	.78

Table 1.1: Quotation Identification and Author Association performance

The remainder part of this dissertation has the following structure:

PUC-Rio - Certificação Digital Nº 1421603/CA

– Chapter 2 introduces the Quotation Extraction task.

– Chapter 3 describes the creation of the corpus QuoTrees 1.0.

– Chapter 4 shows the algorithms used in this work.

– Chapter 5 contains an explanation of our model.

– Chapter 6 gives a description of our experiments.

– Chapter 7 summarizes the dissertation and suggests future research directions.

PUC-Rio - Certificação Digital Nº 1421603/CA

2
Quotation Extraction

In this chapter, we define quotation as well as the quotation types. We begin with the definition of attribution relation, of which quotation is a particular case. We present the Quotation Extraction task and, finally, the differences between our work and the others.

2.1
Attribution Relation

2.1.1
Definition

Pareti [9] defines Attribution Relation (AR) in a text as ascribing the ownership of an attitude towards some linguistic material (i.e. the text itself, a portion of it or their semantic content) to an *entity*. This ownership is expressed by explicitly inserting the agent or experiencer holding the intellectual property of the linguistic material, which can express an assertion or a mental state such as an opinion, a will or some knowledge. In other words, attribution indicates who has expressed some information, with what stance towards it. Figure 2.1 shows some examples of AR.

1. Dr. Smith **said**, "*There is no correlation between smoking cigarettes and lung cancer.*"
2. Some members of the huge crowd **shouted** "*Viva peace, viva*".
3. Mr. Lawson **'s promise** *that rates will be pushed higher if necessary.*
4. "*The Caterpillar people aren't too happy when they see their equipment used like that,*" **shrugs** Mr.George. "*They figure it's not a very good advert*".

Figure 2.1: Examples of Attribution Relations

PUC-Rio - Certificação Digital Nº 1421603/CA

2.1.2 The Elements of Attribution

Pareti [10] deconstructs AR into three main elements:

- *content*, i.e. the attributed material
- source, i.e. the entity responsible for the content
- **cue**, i.e. the lexical anchor linking them

Figure 2.1 highlights these elements. The *content* can range from a single word to multiple sentences (Ex.(4)). The source can be expressed by a named (Ex.(4)) or unnamed entity (Ex.(2)), animate or inanimate entity, or it can be left implicit. The **cue** can be a speech verb (Ex.(1)), another verb (Ex.(4)), a noun (Ex.(3)), an adjective, a preposition, an adverb or just punctuation.

Identifying these three elements provides the insertion of a third part who "owns" the attributed material. For example, in Figure 2.1, it is an utterance (Ex.(2)), a belief or knowledge or an intention (Ex.(3)). However, this variety of expressions encoding leads attribution to a complexity level that makes the definition of a predictive structure not viable.

2.2 Quotation

2.2.1 Towards a Quotation Definition

Defining what quotations are is a trivial task, so trivial that it is not at all easy. Many philosophers tried to define quotation as a linguistic phenomenon, even creating an entire theory to explain it [11–14]. Using the most recent studies on the subject [7, 15, 16], we define quotation as a subset of attribution relation, where the **cue** component is a speech verb.

Why do we restrict quotation to a certain kind of attribution? As we show in section 3.3, our focus is on reported speech. The use of reported speech is crucial in certain contexts, such as the journalistic discourse. Bergler et al. [17] found that there are pieces of new in which over 90% of the sentences include a quotation.

We use the terms quotation/content and author/source/speaker interchangeably.

PUC-Rio - Certificação Digital Nº 1421603/CA

2.2.2 What Counts as a Quotation?

Freitas et al. [18] claims that the identification of reported speech is not unequivocal. The unclear cases typically relate to:

1. The use of conditional as hedge: Eu diria que... ('I would say that...'): *Hoje* **diria** *que há um movimento que se gera a partir do Me e da movimentação de base que existe no Técnico.*

2. The unactualization of reported speech, as in *se disserem* ('if they say...'), *que diga, dirá?* ('will he say?'). In all these cases the actual saying is not presupposed: *Só se tem efeito formativo real e se consegue colocar know how na prática profissional das pessoas se* **disser** *que tem aqui uma actividade...*

3. The use of the modal poder ('can') as hedge: *Pode-se dizer* ('it can be said') and then saying it: **Poderíamos dizer** *que foi mais tempo Ministro que qualquer outro político no pós-25 de Abril.*

4. The frequent omission of the **sayer** but not of what was said: **Dizia-se** *que os estudantes tinham enlouquecido e só faziam coisas aberrantes.*

5. The presence of a report verb in the inflected in the 1st person, present tense: Are we reporting when we say "I say that and that" or is this the actual saying, not reporting? *Hoje* **digo** *que a culpa foi minha.*

6. The presence of negation: *Há quem o veja como candidato presidencial e ele* **nunca disse** *que não.*

7. Nominalization of speech, like in *ele* **falou** *da sua promoção*, that could have been uttered to report a *Fui promovido!*. It is somehow reporting, but more condensed. That the boundaries can be blurred is obvious in the following example, that illustrates a kind of mixed quotation which is hard to identify because there are no indirect speech markers at all. *Em entrevista telefônica na TVI 24, o atual comentador* **falou** *de um homem com "uma inteligência vastíssima", com grande "empenhamento na ação cívica" e um "incansavel combatente da ignorância".*

In this work, we do not consider the presence of reported speech in 1, 2 and 3, because they belong to future tenses. In these cases, we assume that nothing was really said. Pareti [16] consider them as quotations, although she labels these cases as *non-factual*.

PUC-Rio - Certificação Digital Nº 1421603/CA

2.2.3 Types of Quotation

There are three varieties of quotation: direct, indirect and mixed [13]. Figure 2.2 illustrates them: suppose Alice utters 1. She can be properly quoted by any of the sentences 2 to 4.

1. Life is difficult to understand.
2. Alice **said**, "*Life is difficult to understand*".
3. Alice **said** *that life is difficult to understand.*
4. Alice **said** that life "*is difficult to understand*".

Figure 2.2: Types of Quotation

Figure 2.2, Ex.(2) quotes Alice by mentioning the words she uttered. This is direct quotation. This type of quotation is denoted by quotation marks (i.e, "...", '...', ≪ ... ≫).

Ex.(3) is an example of indirect quotation: it quotes her, but could be true even if Alice never uttered any of these words.

Finally, Ex.(4) quotes Alice by reporting what she said, but attributes to her only an utterance of "is difficult to understand". This is called mixed quotation.

2.2.4 A Note About Quotations in the News

Although the main goal of this work is associating quotations with their authors automatically, it is important to frame quotations through a factual perspective. Our intuition, at first glance, may consider the content of direct quotations more reliable than indirect ones, as the former reproduces exactly what the source says. However, we must be really careful with this claim.

Harry [19] says journalists use direct quotation to reproduce what others say, and varieties of indirect quotation to boil down or more distantly represent what others say. In either case the quoting reporter engages in a two-step compositional process, using language that, from a linguistic perspective, constructs verbal (i.e., linguistic) signs interpretable as the immediate 'voice' or the more distant 're-voicing' of what someone else supposedly said. But all quotation, from direct to indirect, is a re-voicing. To quote someone is always to re-voice previous speech at some temporal distance from its original utterance.

PUC-Rio - Certificação Digital Nº 1421603/CA

Taking this into consideration, we need a reliable source to be considered trustworthy. Ascribing the content to a major newspaper is more effective than directly citing an unknown journalist.

Besides, the focus of this work is on news not only because of the ubiquity of this phenomenon in the news genre, but also because one of the goals of collecting such resource is to enable studying the effect of attribution on information.

2.3 The Task

The purpose of the Quotation Extraction task is to identify direct and indirect quotations in a text and associate them with their sources. We divide this task into three subtasks: content identification, source identification and association between content and source.

We show an example of the content identification subtask in Figure 2.3. Quotations are in *italics*.

> O governador alegou *que, se abandonassse a candidatura, poderia se tornar refém de Quércia*. A declaração de Fleury de que continua candidato não convenceu alguns prefeitos que o acompanhavam. «*Ele já está abraçando a candidatura Quércia*», disse Itamar Borges (PMDB), de Santa Fé do Sul. Fleury aproveitou a viagem para responder ao ex-secretário José Machado de Campos Filho, que o chamara de traidor por não apoiar Quércia. «*Para quem quer ser candidato ao Senado, faltou bom senso e juízo*», disse.

Figure 2.3: Quotation identification subtask

In the source identification subtask, we list the possible authors. Figure 2.4 presents an example in which the candidates are in **bold** and tagged with an integer subscript.

> **O governador**$_1$ alegou que, se abandonassse a candidatura, poderia se tornar refém de Quércia. A declaração de Fleury de que continua candidato não convenceu alguns prefeitos que o acompanhavam. «Ele já está abraçando a candidatura Quércia», disse **Itamar Borges (PMDB), de Santa Fé do Sul**$_2$. **Fleury**$_3$ aproveitou a viagem para responder ao ex-secretário José Machado de Campos Filho, que o chamara de traidor por não apoiar Quércia. «Para quem quer ser candidato ao Senado, faltou bom senso e juízo», disse.

Figure 2.4: Author candidates subtask.

PUC-Rio - Certificação Digital Nº 1421603/CA

Finally, in the association subtask, we associate the content with its source. We show an example of three associations between content and source in Figure 2.5. Each content is tagged with its respective source label.

> **O governador**$_1$ alegou *que, se abandonassse a candidatura, poderia se tornar refém de Quércia*$_1$. A declaração de Fleury de que continua candidato não convenceu alguns prefeitos que o acompanhavam. «*Ele já está abraçando a candidatura Quércia*»$_2$, disse **Itamar Borges (PMDB), de Santa Fé do Sul**$_2$. **Fleury**$_3$ aproveitou a viagem para responder ao ex-secretário José Machado de Campos Filho, que o chamara de traidor por não apoiar Quércia. «*Para quem quer ser candidato ao Senado, faltou bom senso e juízo*»$_3$, disse.

Figure 2.5: Association between content and source subtask

For the quotation and source identification subtasks, we add the synctatic dependecy annotation in the corpus. For the association between content and source subtask, we provide the input corpus with content and source candidates annotations.

We might think this task is easily solved by a rule based system, as quotations usually follow a specific formation pattern. For example, direct quotations are always between quotation marks. However, there are many situations in which these marks are not used to denote quotations. Figure 2.6 presents some examples in which the "pseudo" quotations are in *italics*.

1. Desde o último dia 13, «*Confissões de Adolescente*» pode ser vista pelos teens portugueses.
2. Essas mudanças geralmente coincidem com a «*crise da meia idade*», que é definida não pela faixa etária, mas como um certo momento em que as pessoas reavaliam as escolhas feitas na sua vida.
3. «*A Interpretação dos Sonhos*», um dos textos fundadores da psicanálise, apresenta toda uma tropologia do ato de sonhar.
4. A surpresa descrita por Pedro Collor atingiu todos os partidos que integram a chamada «*Nova Força*» da política local, liderada por Lessa.

Figure 2.6: Some examples in which quotation marks are not used to denote quotations

Besides, our proposal is to tackle the indirect quotations too. For these, it is not trivial to build such a system, as they do not have any denote marks.

PUC-Rio - Certificação Digital Nº 1421603/CA

In addition, some verbs can be used as speech verbs, although their meaning changes by context. Figure 2.7 shows some examples.

1. O governador **acrescentou** que pedirá ao Nosso Senhor do Bonfim para que o plano dê certo. / Segundo Galan, os novos produtos e serviços **acrescentam** 10% ao faturamento, que foi de US$ 360 milhões em 93.

2. «No último jogo, nós teremos um prazer, ou de sermos campeões ou de dar o título ao São Paulo», **garantiu** Moacir. / Manoel Carlos Marques Beato, do restaurante Fasano e José Sebastião Figueiredo, sommelier e proprietário do La Bicocca, **garantiram** suas vagas no Concurso Brasileiro de Sommeliers.

3. Nos últimos cinco anos, só na Argentina, a Coca-Cola investiu US$ 800 milhões na aquisição de novos equipamentos e desenvolvimento de novas tecnologias, **lembrou** Ivester após encontro com o presidente argentino Carlos Menem. / As construções **lembram** vagamente uma verdadeira vila de pescadores.

Figure 2.7: Some examples in which verbs are used as speech verbs or not

2.4 Related Studies

Research and commercial systems for the automatic identification and extraction of quotations have multiplied in recent years. The NewsExplorer system extracts quotations from multilingual news [20]. It uses lists of verbs of speech (e.g. said, commented), quotation marks (e.g. ' "), general modifiers (e.g. yesterday), determiners (e.g. the) and people's names. Then, it uses regular expressions to identify quotations and to associate them with their authors. The NewsExplorer just identifies direct speech quotations.

Krestel et al. [21] developed a quotation extraction and attribution system that combines a lexicon of 53 common reporting verbs and his own grammatical rules to detect constructions that match 6 general lexical patterns. They evaluate their work on 7 articles from the Wall Street Journal, which contain 133 quotations, achieving macro-averaged Precision of 99% and Recall of 74% for quotation span detection. Although their results are high, the number of examples is very low to have statistical validity. Their system tackles direct and indirect quotations.

PUC-Rio - Certificação Digital Nº 1421603/CA

The PICTOR system, by Schneider et al., relies instead on a context-free grammar for the extraction and attribution of direct and indirect quotations [22]. PICTOR yielded 75% and 86% in terms of words correctly ascribed to a quotation or speaker, while it achieved 56% and 52% when measured in terms of completely correct quotation-speaker pairs.

The SAPIENS system, by de La Clergerie et al., extracts quotations from French news, by using a lexicon of reporting verbs and syntactic patterns to extract the complement of a reporting verb as the quotation span and its subject as the source [23]. They evaluated 40 randomly sampled quotations and found that their system made 32 predictions and correctly identified the span in 28 of the 40 cases. SAPIENS considers direct and indirect quotes.

The Verbatim system, by Sarmento and Nunes, extracts quotations from Portuguese news feeds by first finding one of 35 speech verbs and then matching the sentence to one of 19 patterns, that include direct and indirect quotations [24]. Their manual evaluation shows that 11.9% of the quotations Verbatim finds are errors and that the system identifies approximately one distinct quotation for every 46 news articles.

The system presented by Fernandes [6] also works on Portuguese news. His work introduces GloboQuotes, a corpus of 685 news items containing 1,007 quotations of which 802 were used to train a Structured Perceptron model [**?**]. The overall system achieves $F1$ of 76.8%. His work does not tackle indirect quotations.

Pareti et al. [7] proposes a supervised learning system to extract direct, indirect and mixed quotations. They use a linear chain Conditional Random Field (CRF) as the learning algorithm and apply their approach on two datasets. For the quotation identification task only, without capturing the speaker, they achieve $F1$ of 59% and 60% for indirect quotations and $F1$ of 73% and 78% for all quotations.

Our proposal extends the work [6] and is the first work to address direct and indirect quotations in Portuguese using a Machine Learning approach.

PUC-Rio - Certificação Digital Nº 1421603/CA

3
A Corpus for Direct and Indirect Quotation

The machine learning approach is based on the assumption that a computer is able to learn through examples. In order to build an artificial intelligence that identifies quotations, we need to feed it with a large amount of examples of quotations. For this purpose, we build a dataset [1] that contains several examples of indirect and direct quotations.

We take several hundred news items and annotate all the quotations in the text. The annotation of the corpus is recognized as time-consuming work. This is because, for a dataset to be considered good, aside from needing to contain many and varied examples of the phenomenon that we want the machine to learn, the annotations need to be consistent and free of errors. A common way of optimizing this process is to first have the computer do the annotations automatically and then have a specialist check them manually. This is the strategy we follow in this work.

This chapter describes these two steps in detail. The results are the development of a new schema for the annotation of quotations in Portuguese and the dataset itself. We start by presenting the lexico-syntactic patterns used to annotate the quotations automatically. For the second step, we discuss the guidelines for tackling some particularities of Portuguese. Finally, we present QuoTrees 1.0, the first corpus in Portuguese fully annotated with direct and indirect quotations.

PUC-Rio - Certificação Digital Nº 1421603/CA

3.1
The Sources

Instead of building a corpus from a raw text, we use two public datasets that are already tokenized and annotated with linguistic labels: *Bosque* and *Floresta Virgem*. Both are part of *Floresta Sintá(c)tica* [25]: a collection of documents in Portuguese annotated automatically by the parser *PALAVRAS* [26].

The two datasets contain news items from two sources: Folha de São Paulo and Público. Both are daily newspapers: the first, Brazilian and the

[1] In Linguistics, this kind of dataset is called *annotated corpus*. It is made of raw texts enriched with linguistic information. In general, this kind of dataset is used in Natural Language Processing tasks.

latter, Portuguese. The new items from Folha are from 1994 and those of Público are from 1991 to 1998 [27].

Although the sources are the same, the sentences in *Bosque* and *Floresta Virgem* are unique to each. *Bosque* is a subset of Floresta Virgem, fully revised and corrected by linguists with a current size of 9,368 sentences. *Floresta Virgem* is a set of trees automatically created from the output of the PALAVRAS parser, corresponding roughly to the first million tokens of the CETEMPúblico and CETENFolha corpora.

In this work, we use these datasets in the *AD* (Árvores Deitadas - *phrase structure tree*) format [28], that is easier for human revision (section 3.4 shows an example of this format). We use only the first 37,386 sentences of *Floresta Virgem*, an amount feasible to be manually revised without compromising the automatic learning process/algorithm.

3.2 Reported Speech in Portuguese

Freitas [29] developed a glossary of verbs in Portuguese that introduces reported speech. She conducted a corpus-based descriptive research, analysing both monolingual corpora as well as translations of the verb *to say* from English to Portuguese. Her work is motivated by the request for translating dialogues of reported speech, without constantly using the verb *dizer* in Portuguese. As a side-effect of the glossary, Freitas [29] identifies eight syntactic patterns in which reported speech verbs are commonly found.

Based on these patterns, we create five syntactic rules for the automatic phase of the annotation process. In the next section, we show the five rules. In the section 3.4, we describe how we use these rules to build the annotator.

3.3 The Rules

We use the following abbreviations to describe the implementation of the rules:

- *ACC*: the quotation. This refers to *direct objects*.
- **VSAY**: reported speech verb.
- SUBJ: the speaker of quotation. This refers to the *subject* of the sentence.
- quotation mark: It appears as “ . . . ” or « . . . »

Each rule encompasses a specific list of reported speech verbs. We show these lists in appendix A.

PUC-Rio - Certificação Digital Nº 1421603/CA

In addition, each rule is valid if the reported speech verb is in the past or present tense, as discussed in section 2.2.2.

Rule 1. *Quotation + speech verb + subject*

ACC», **VSAY** SUBJ *ACC*», SUBJ **VSAY**

Here are some sentences that are in accordance with this rule:

(a) «*Ninguém da minha família participa de sequestro*», **disse** Silva.

(b) «*Tudo que você escreve vende, não é mesmo?*», **perguntou** Larry.

(c) «*É mais do que isso*», **respondeu** Gabus Mendes.

According to [29], this may be the most common pattern in Portuguese. It is the first rule that captures direct quotations, since it takes the quotation marks into consideration.

Rule 2. *Subject + speech verb + Quotation*

SUBJ **VSAY**: «*ACC* **VSAY** SUBJ: «*ACC*

Some examples of sentences that are in accordance with Rule 2:

(a) Zico, em conversa exclusiva com a Folha, **disse**: «*Estou muito contente com essa homenagem*».

(b) O ministro da Educação, Murílio Hingel, **admite**: «*Apesar de todos os esforços, nosso ensino básico é vexaminoso.*»

(c) Ele **continua**: «*Aqui eu acho que eu retomo a urgência, de um novo ângulo.*»

Rule 2 is another pattern that captures direct quotations.

Rule 3. *Subject + speech verb +* que *(that) + Quotation*

SUBJ **VSAY** que *ACC*

Next, we show some sentences that are in accordance with Rule 3:

PUC-Rio - Certificação Digital Nº 1421603/CA

(a) Sobre a lateral esquerda, Parreira **disse** *que quem deve jogar amanhã é Leonardo.*

(b) Genro, em tom duro também, **respondeu** *que aquela não era uma visão de um dirigente de expressão nacional como Dirceu.*

(c) O teólogo católico **admite**, no entanto, *que o tema das mulheres é um problema real, ao qual a Igreja Católica também tem que dar resposta.*

This rule is the only one that captures indirect quotations. Despite its simple and rigid structure, it is also very frequent in Portuguese [29].

The last two rules reflect quotations that are separated by a speech verb. They are the rarest patterns, with Rule 5 being more common in literary texts.

Rule 4. *Part of quotation + speech verb + subject + part of quotation (with quotation marks)*

ACC», **VSAY** SUBJ, «*ACC*
ACC», SUBJ **VSAY**, «*ACC*

(a) «*Vossa Santidade*», **respondi**, «*não só eu estou contente, todos nós estamos muito contentes*».

(b) «*Sem dúvida alguma, ele é o melhor boxeador do mundo!*», **admite**, «*mas enquanto ele continuar dizendo que é o melhor lutador, estou pronto a desafiá-lo*».

Rule 5. *Part of quotation + speech verb + subject + part of quotation (without quotation marks)*

ACC, **VSAY** SUBJ, *ACC*
ACC, SUBJ **VSAY**, *ACC*

ACC - **VSAY** SUBJ - *ACC*
ACC - SUBJ **VSAY** - *ACC*

(a) *O contrato*, **diz** ele, *expirou ano passado e só foi renovado em fevereiro.*

(b) *Para o ministro*, **disse** Simon, *as mudanças feitas são assimiláveis pelo plano.*

PUC-Rio - Certificação Digital Nº 1421603/CA

(c) *Essas empresas*, **diz** Arruda, *reconheceram ter emitido «notas frias», sem o recebimento de nenhum serviço da empresa do pianista João Carlos Martins e de seu sócio Ettore Gagliardi.*

3.4 Using the Rules

Based on the rules of section 3.3, we create an automatic annotator that processes *Bosque* and part of *Floresta Virgem* (described in section 3.1). We show an example of how this annotator works. Here is an excerpt of *Bosque*:

```
SOURCE: CETENFolha n=55 cad="Mundo" sec="pol" sem="94a"
CF55-6 O secretário do Tesouro, Lloyd Bentsen, disse que a questão
ética vai ser examinada agora.
A1
STA:fcl
=SUBJ:np
==>N:art('o' <artd> M S) O
==H:n('secretário' <np-def> M S) secretário
==N<:pp
===H:prp('de' <sam->) de
===P<:np
====>N:art('o' <-sam> <artd> M S) o
====H:n('tesouro' <np-def> <prop> M S) Tesouro
====,
====APP:np
=====H:prop('Lloyd_Bentsen' M S) Lloyd_Bentsen
====,
=P:vp
==MV:v-fin('dizer' PS 3S IND) disse
=ACC:fcl
==SUB:conj-s('que') que
==SUBJ:np
===>N:art('o' <artd> F S) a
===H:n('questão' <np-def> F S) questão
===N<:adjp
====H:adj('ético' F S) ética
==P:vp
===AUX:v-fin('ir' PR 3S IND) vai
===AUX:v-inf('ser') ser
===MV:v-pcp('examinar' <icl-subst> <passive> F S) examinada
==ADVL:advp
```

PUC-Rio - Certificação Digital Nº 1421603/CA

```
32 ===H:adv('agora' <kc>) agora
33 =.
```

This sentence can be represented by the dependency tree shown in Figure 3.1 (we omit some labels for clarity):

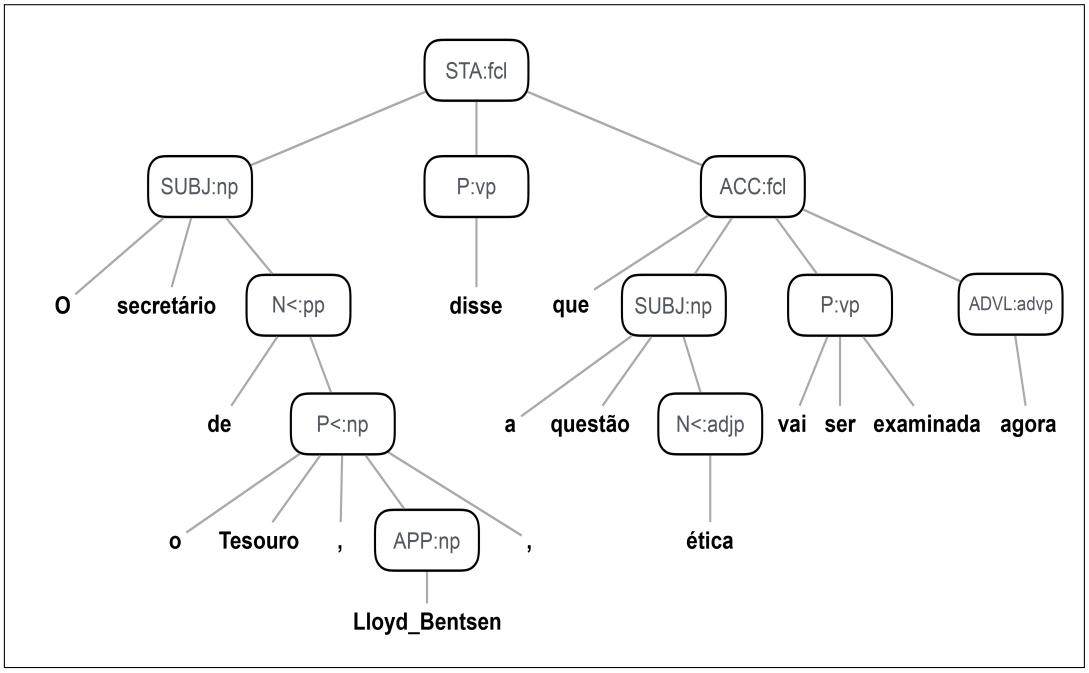

Figure 3.1: Dependency tree based on *árvores deitadas* format

We assume that each element of a rule is on the same level in the tree and has the same parent node. In other words, they are siblings. In the *AD* format, the level is given by the number of equal symbols (=) that precedes a node. For instance, in line 8, the level of the token `secretário` is 2, because there are two = preceding it. It is a child of node `SUBJ:np` (in line 6) and has two siblings: the nodes of lines 7 and 9.

Looking back at Figure 3.1, this sentence belongs to Rule 3: lines 6, 18 and 20 are, respectively, SUBJ, **VSAY** and *ACC*. They are on the same level of the tree and have the same parent node (at line 5). Also, we can verify that the verb *dizer* (to say) is a speech verb and the conjunction *que* (that) appears inside the ACC node.

After checking the rule that the sentence belongs to, we annotate the speaker and the quotation. In order to retrieve each element, we have to get all the descendants of the node, in the order they appear in the file, removing the labels-only nodes in between. For example, to get the text of the SUBJ of the sentence, from line 6 through line 17, we have:

PUC-Rio - Certificação Digital Nº 1421603/CA

O secretário de o Tesouro, Lloyd_Bentsen,

We proceed in the same way to get the quotation. Once we have identified these elements, we can instruct the program to annotate them.

3.4.1 Why Do We Need a Machine Learning Model?

If we are able to build such a ruled-based program, why do we still need a Machine Learning model? The purpose of developing this rule-based system is to speed up the annotation phase.

However, suppose that we want to use such a system in a production environment. The first obstacle is had the text parsed by PALAVRAS, which is a paid tool. Moreover, suppose that we have unlimited access to PALAVRAS. In this case, with such system, one could easily extract the quotations that adhere to the rules. Either, one could build a corpus for quotation using different sources in Portuguese beyond *Bosque* and *Floresta*, with a much larger content.

Nevertheless, suppose that we want to extend our system, teaching it to retrieve quotations that do not adhere to either of these rules. For example, the following sentence has a quotation, although it does not conformed to any rule:

Caso a opção seja pelo aparelho multiuso, o comprador deve checar se o produto tem assistência técnica, **diz** ela.

Instead of searching for new patterns and verbs that are rare and difficult to formalize, we could manually annotate these sentences in the *corpus*. We show some examples of this type in section 3.5.1.

Working on the *corpus* allows the ML model to learn these patterns without having to describe them. These features of flexibility and ability to generalize are part of its beauty.

3.5 Generating the Dataset

QuoTrees 1.0 is composed of several *feeds*. A *feed* is a group of sentences from the same news item.

The dataset has five columns: token, part of speech (POS), simplified dependency, author and quotation. The columns are separated by a *tab* character.

PUC-Rio - Certificação Digital Nº 1421603/CA

We use the excerpt of *Bosque* showed in section 3.4 to illustrate the generation of the column that follows.

To generate the first column, we leave just the lines with valid tokens. A line has a valid token if it contains a piece of the sentence at the end (and not just a label after the =). As a result, we use the same tokenization provided by the sources (*Bosque* and *Floresta Virgem*). In the excerpt, lines 7, 8 and 10 are examples of lines with valid tokens. Lines 6, 9 and 11 are examples of lines that are discarded.

The second column, part of speech (*POS*), is also taken from the sources: in the token line, it is the first label after the colon (:). In the excerpt, line 7 has a *POS* of `art`, because it is the first label after the colon in `>N:art`.

We explain the generation of the third column in the section 3.5.2.

The fourth column has the index of the author. The same index can appear at column five, linking the author with the quotation. Table 3.1 shows an example.

Token	**Dilma**	**disse**	**que**	**não**	**renuncia**
Author	1	-	-	-	-
Quotation	-	-	1	1	1

Table 3.1: Linking author to quotation through index

The author index is incremented each time a new quotation appears in a feed. The index for the first author/quotation is 0 and we reset it at the beginning of a new feed.

3.5.1 Linguistic Decisions

In this section, we discuss how we handle some problems that arise during the application of the rules and extraction of quotations.

1. **Mixed quotations**: In the case where there is a quotation inside another, we mark the outermost. For example: Prandi **disse** ainda *que a empresa está elaborando «normas factíveis de serem executadas para a solução ou minimização dos problemas» existentes no local.*

2. **Pronominal Anaphora**: Sometimes, the rules capture quotations linked to pronouns. For instance: Ele **afirma** *que os candidatos não podem apresentar «patologia com perspectiva presente, de incapacidade no futuro»*. In these cases, we annotate Ele as the speaker. Ideally, we should link all the quotations to a more clearly-identified speaker. However, in QuoTrees 1.0, we find only 30 occurrences of pronouns as annotated

PUC-Rio - Certificação Digital Nº 1421603/CA

speakers. In addition, the ML task related to identifying the real entity of a pronoun is coreference resolution, which does not have good quality results [30, 31].

3. **Missing subject**: In Portuguese, the subject is not always explicit in the text. We can infer it by the morphology of the verb and the context. Since the rules only identify the author in the sentence of the quotation, we adapt them to identify the quotes of missing subjects. In these cases, we annotate the quotation column with -1. For example: «*Em 79 houve uma leve reabertura de crédito, mas nada significativo*», afirma.

3.5.2 Simplified Dependency

As we see in section 2.3, we need to retrieve the candidates of quotations and authors of a sentence. To do that, we build the *simplified dependency* column. Table 3.2 shows an example of it.

PUC-Rio - Certificação Digital Nº 1421603/CA

Id	Token	POS	S. Dependency
1	O	ART	ChildL1:Root1
2	secretário	N	ChildL1:Root1
3	de	PRP	ChildL1:Root1
4	o	ART	ChildL1:Root1
5	Tesouro	N	ChildL1:Root1
6	,		ChildL1:Root1
7	Lloyd_Bentsen	PROP	ChildL1:Root1
8	,		ChildL1:Root1
9	disse	FIN	Root1
10	que	S	ChildR1:Root1
11	a	ART	ChildR1:Root1
12	questão	N	ChildR1:Root1
13	ética	ADJ	ChildR1:Root1
14	vai	FIN	ChildR1:Root1
15	ser	INF	ChildR1:Root1
16	examinada	PCP	ChildR1:Root1
17	agora	ADV	ChildR1:Root1

Table 3.2: Simplified dependency labeling example

To generate this column, for each reported speech verb we:

1. Mark it with the label `Root`i
2. Mark its left siblings nodes with the label `ChildL`j`:Root`i
3. Mark is right siblings nodes with the label `ChildR`k`:Root`i

In these steps, i is the index of the speech verb in the feed. We assign 1 for the first speech verb, 2 for the second etc. j is the index of the left sibling of the speech verb node. For example, if the speech verb node has two left siblings, the first has index 1 and the second, 2. It is important to note that the index is propagated to all the descendants of the sibling node. The k is generated in the same way, but for the siblings of the right side.

We generate this column because the sources have both the information of POS and dependency. We identify that a node is a speech verb if its POS is *verb* and it is in the list of speech verbs. The dependency information gives us the siblings of a node.

For a production version, generating this column is a straightforward process, with a dependency parser and a *POS* tagger. Both are ML tasks with good results for Portuguese [32, 33].

3.6 Statistics

Table 3.3 summarizes the *corpus* statistics.

#Tokens	*#Sentences*	*#Feeds*	*#Quotations*
460,966	24,139	5,199	1,663

Table 3.3: Statistics of QuoTrees 1.0

We also count the number of quotations *per rule*. Table 3.4 shows these results. We divide the counting in *quotations with subject* and *quotations without subject*, the later being the sentences with missing subject. In these two cases, the number of indirect quotations is bigger than direct quotations.

Quotations' Type	*Direct*					*Indirect*
Rule	*1*	*2*	*4*	*5*	*Total*	*3*
# SUBJ	513	2	0	18	533	804
# Without SUBJ	117	1	0	0	118	208

Table 3.4: Quotations by rule and type in QuoTrees 1.0

PUC-Rio - Certificação Digital Nº 1421603/CA

4
Algorithms

In this chapter, we show the algorithms used to build the ML model. We start by an overview of the ML algorithms and their categories. Next, we define the *perceptron* and how its *learning algorithm* was adapted to deal with structures. Finally, we present the Weighted Interval Scheduling algorithm.

4.1
Learning Process

Machine Learning algorithms allow computers to learn from data. Given a bunch of data, the result of the learning algorithm is a *model*. Strictly speaking, it is a mathematical model of a function, but we can think of it as a model of a human *specialist*, who now has information about the data. Based on this information, our specialist "knows" how to predict an example she has never seen.

The learning process is divided into two main categories: *supervised* learning and *unsupervised* learning. The first, also referred to as learning with a teacher, consists of giving examples with "correct answers" to our model. For instance, suppose we are trying to build a model that predicts apartment prices. In the dataset, each example could include many characteristics of real apartments (such as size, neighborhood etc.) and its price. The price would be the "correct answer".

In unsupervised learning models, there are no "correct answers". Instead, their goal is to find patterns.

The Perceptron's Algorithm, used in this work, was the first proposed model of *supervised* learning [34]. Before we show it, we need to define what is a *perceptron*.

4.2
The Perceptron

PUC-Rio - Certificação Digital Nº 1421603/CA

4.2.1 Definition

A *perceptron* [34] is a mathematical representation of a human brain neuron. Figure 4.1 shows an example of a perceptron. It is composed by a linear combination of its inputs and weights, an activation function and an output.

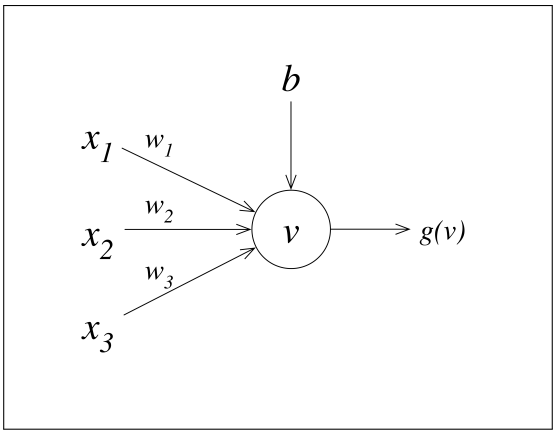

Figure 4.1: An Example of Perceptron

We denote the inputs by the vector $\boldsymbol{x} = [x_1, x_2, \ldots x_m]$, the weights by $\boldsymbol{w} = [w_1, w_2, \ldots w_m]$ and the externally applied bias by b. The summing node of the neural model is given by

$$v = \sum_{i=1}^{m} w_i x_i + b \tag{4-1}$$

If we treat bias b as a component of $\boldsymbol{w}$, $w_0 = b$, and $x_0 = 1$, we define

$$v = \sum_{i=0}^{m} w_i x_i = \boldsymbol{w} \cdot \boldsymbol{x} \tag{4-2}$$

Finally, we define the *activation function* g by

$$g(v) = \begin{cases} \text{-1} & \text{when } v < 0 \\ 1 & \text{when } v \geq 0 \end{cases}$$

In this example, we define g as signal function, but other functions can be used [35,36]. The result of the activation function is the perceptron's *output*. In the same way a human brain neuron passes or does not pass along an electrical charge to its neighbors, a perceptron can be *firing* (when the output is 1) or *not firing* (when the output is -1). These two states allow the perceptron to perform pattern classification with two classes.

PUC-Rio - Certificação Digital Nº 1421603/CA

4.2.2 Learning Algorithm

The breakthrough of the perceptron model, when it was first proposed by Rosenblatt [34], is that it consists of a single neuron with *adjustable* synaptic weights and bias. The Perceptron's Algorithm is used to adjust these free parameters of the model. Figure 4.2 presents the algorithm.

The learning consists of initializing the weight vector with zeros and predicting the result of this classification. If the result is correct, we continue the process. If the prediction does not correspond to the desired output, the weight vector is updated by adding or subtracting the feature vector of the example, based on the correct output. These steps are repeated several times for fine-tuning the weight vector.

Input: $D \leftarrow \{(\boldsymbol{x}, y)\}$ binary data set
Output: $\boldsymbol{w}$
1: **while** no convergence **do**
2: $\quad \boldsymbol{w} \leftarrow 0$
3: $\quad$ **for each** $(\boldsymbol{x}_i, y_i)$ **do**
4: $\quad\quad \hat{y}_i \leftarrow sign(\boldsymbol{w} \cdot \boldsymbol{x}_i)$
5: $\quad\quad$ **if** $y_i \neq \hat{y}_i$ **then**
6: $\quad\quad\quad \boldsymbol{w} \leftarrow \boldsymbol{w} + y_i \boldsymbol{x}_i$
7: **return** $\boldsymbol{w}$

Figure 4.2: Perceptron Learning Algorithm

Rosenblatt proved that if the examples used to train the perceptron are drawn from two linearly separable classes, then the perceptron algorithm converges. In this case, the algorithm positions the decision surface in the form of a hiperplane between the two classes. This proof is known as the *perceptron convergence theorem*.

If the perceptron is able only to classify examples between two classes, how can it be used to predict structured data, such as in the Quotation Extraction task? In the next section, we investigate a generalization of the Perceptron algorithm, the *Structure Perceptron algorithm*, that is able to handle this kind of problem.

4.3 Structured Perceptron

In 2002, Collins [4] proposed a modification of the proof of convergence of the Perceptron algorithm, that later was known as the *Structured Perceptron algorithm*.

PUC-Rio - Certificação Digital Nº 1421603/CA

Given a training set D, composed of pairs $(\boldsymbol{x}, \boldsymbol{y})$ of correct inputs-outputs, both with complex structures, the algorithm updates model $\boldsymbol{w}$ in order to improve its quality. At each interation, a training instance $(\boldsymbol{x}, \boldsymbol{y})$ is drawn from D and two major steps are performed: prediction using the current model and model update based on the difference between the correct and the predicted outputs. Figure 4.3 shows the algorithm.

Input: $D \leftarrow \{(\boldsymbol{x}, \boldsymbol{y})\}$
Output: $\boldsymbol{w}$
1: **while** no convergence **do**
2: $\quad \boldsymbol{w} \leftarrow 0$
3: $\quad$ **for each** $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ **do**
4: $\quad\quad \hat{y}_i \leftarrow \arg\max_{y' \in \mathcal{Y}(x)}(\boldsymbol{w} \cdot \boldsymbol{\Phi}(\boldsymbol{x}_i, \boldsymbol{y'}))$
5: $\quad\quad \boldsymbol{w} \leftarrow \boldsymbol{w} + \boldsymbol{\Phi}(\boldsymbol{x}_i, \boldsymbol{y}_i) - \boldsymbol{\Phi}(\boldsymbol{x}_i, \boldsymbol{y'})$
6: **return** $\boldsymbol{w}$

Figure 4.3: Structured Perceptron Algorithm

When the current model makes a correct prediction $\hat{y} = y$, the model does not change, that is $w_{t+1} \leftarrow w_t$. When the prediction is wrong, the update rule favors the correct output y over the predicted one $\hat{y}$. Regarding binary feature functions, for instance, the update rule increases the weights of features that are present in y but missing in $\hat{y}$ and decreases the weights of features that are present in $\hat{y}$ but not in y. The weights of features that are present in both y and $\hat{y}$ are not changed.

The challenge to build a Structured Perceptron model is to choose an optimization problem that solves the proposed task and map the examples in input-output pairs in such a way that can be selected by the prediction problem. The optimization problem defines the predictor $F(\boldsymbol{x})$, whose objective is to find a $\hat{y}$ which maximizes the dot product between model $\boldsymbol{w}$ and feature vector $\boldsymbol{\Phi}(\boldsymbol{x}, \boldsymbol{y'})$.

$$F(\boldsymbol{x}) = \text{argmax}_{y' \in \mathcal{Y}(x)}(\boldsymbol{w} \cdot \boldsymbol{\Phi}(\boldsymbol{x}, \boldsymbol{y'}))$$

Thus, the learning problem consists of determining $\boldsymbol{w}$, such that the resulting predictor $F(\boldsymbol{x})$ is accurate on the training data and, moreover, shows good generalization performance on unseen data.

In the Quotation Extraction task, as we see in chapter 5, we map pairs of candidates quotation-author into tasks for the The Weighted Interval Scheduling Algorithm, that we describe in section 4.4.

PUC-Rio - Certificação Digital Nº 1421603/CA

4.3.1 Large Margin Training

The Structure Perceptron algorithm finds a classifier with no concern about its margin. However, it is well known that large margin classifiers provide better generalization performance for unseen data. The MIRA [37] is a generalization of the Structured Perceptron algorithm which generates a large margin classifier. However, in this work, we use another large-margin generalization of the Structured Perceptron that is based on the margin rescaling technique for Structured Support Vector Machines [38]. For a training instance $(\boldsymbol{x}, \boldsymbol{y}) \in D$, instead of the ordinary discriminant problem $F(\boldsymbol{x})$, we use a *loss-augmented* discriminant problem in step (4) of the Structured Perceptron algorithm that measures the difference between the predicted output $\hat{\boldsymbol{y}}$ and the correct output $\boldsymbol{y}$.

$$F_\ell(\boldsymbol{x}) = \text{argmax}_{y' \in \mathcal{Y}(x)}(\boldsymbol{w} \cdot \boldsymbol{\Phi}(\boldsymbol{x}, \boldsymbol{y'}))$$

where $\ell(.,.) \geq 0$ is a given loss function that measures the difference between $\hat{\boldsymbol{y}}$ and $\boldsymbol{y}$

4.4 The Weighted Interval Scheduling Algorithm

The Weighted Interval Scheduling (WIS) is a well-known optimization problem. We have a set of n tasks which have a start time s_i, an end time e_i and a weight w_i. We want to find the maximum-weight subset of non-overlapping tasks.

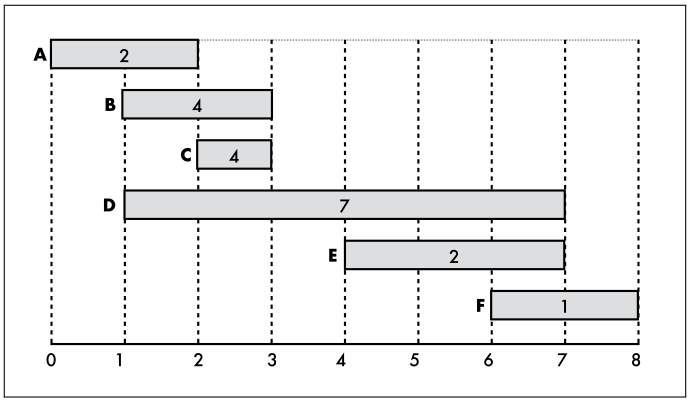

Figure 4.4: An Example of WIS problem

Figure 4.4 shows an instance of the problem. The tasks A to F are distributed along the time horizontal axis with their weights values inside each task's rectangle. For example, task A is defined as ($s_A = 0$, $e_A = 2$, $w_A = 2$);

PUC-Rio - Certificação Digital Nº 1421603/CA

task $B = (s_B = 1, e_B = 3, w_B = 4)$. We want to find a subset of tasks that the sum of their weights is maximum, but they can't overlap each other in time. In this example, the subset is $S = \{A, C, E\}$ and the sum of their weights is 8.

Input: $T \leftarrow \{(s, e, w)\}$ tasks data set
Output: a scheduling of weight M of tasks $S \subset T$

1: sort tasks by increasing end times.
2: compute p_i for i from 1 to n.
3: $M_0 \leftarrow 0$
4: $S_0 \leftarrow \{\}$
5: **for** $i = 1$ to n **do**
6: **if** $w_i + M_{p_i} \geq M_{i-1}$ **then**
7: $M_i \leftarrow w_i + M_{p_i}$
8: $S_i \leftarrow \{i\} \cup S_{p_i}$
9: **else**
10: $M_i \leftarrow M_{i-1}$
11: $S_i \leftarrow S_{i-1}$
12: **return** S_n, M_n

Figure 4.5: Linear time algorithm for WIS

The problem is efficiently solved by dynamic programming. Figure 4.5 presents a linear time algorithm for WIS. We define M_i as the maximum weight of any set of compatible tasks, all of which end by e_i. Moreover, we define S_i as the tasks involved in schedule M_i. We also define p_i as the task k with the largest index less than i and $e_k < s_i$. If there is no task that satisfies that condition, $p_i = 0$.

The WIS solution is progressively built: at each iteration i of n, we determine if task i will be part of the solution, or the task $i - 1$. By the end of the iterations, we have the maximum weight of all the compatible tasks in M_n. We also have the tasks involved in the solution in S_n.

PUC-Rio - Certificação Digital Nº 1421603/CA

5
The Machine Learning Model

In this chapter, we explain how to use the corpus created in chapter 3 with the algorithms described in chapter 4 to build the machine learning model that tackles the Quotation Extraction task. We start by describing the preprocessing step that translates the corpus in structures. Next, we present the proposed feature set. Finally, we explain how the input-output mapping is made.

5.1
Preprocessing

In order to train the model and to make a prediction, we need to extract the examples from the corpus built in Chapter 3, in a preprocessing stage.

Each news feed in the corpus generates an input-output pair $(\boldsymbol{x}, \boldsymbol{y})$ example. The input $\boldsymbol{x} = (\boldsymbol{q}, \boldsymbol{a})$ comprises two sets: the candidates quotes $\boldsymbol{q} = \{q_1, \ldots, q_N\}$ and the candidate authors $\boldsymbol{a} = \{a_1, \ldots, a_K\}$. Each quote $q_i = (s_i, e_i)$, for $i \in \{1, \ldots, N\}$, is a segment in the document and is represented by its starting token s_i and its end token e_i, where $s_i \leq e_i$. The output $\boldsymbol{y} = (y_1, \ldots, y_N)$ is a vector of author indexes, where $y_i \in \{1, \ldots, K\}$ indicates the author associated to the quote q_i.

In this way, the first step in the preprocessing stage is searching for candidates quotes and candidates authors in a news feed of the corpus. This is described next.

5.2
Searching for Candidates

To find the quotations candidates and speakers candidates, we use the simplified dependency column of the corpus. Table 5.1 shows an example.

Token	**Dilma**	**disse**	**que**	**não**	**renuncia**
SD	ChildL1	Root1	ChildR1	ChildR1	ChildR1

Table 5.1: Example of Simplified Dependency

For each node labeled as *Rooti*, we permute its children nodes, taken two at a time. Each permutation is an attribution candidate (q_i, a_j), in which

PUC-Rio - Certificação Digital Nº 1421603/CA

the first element is the quotation candidate q_i and the second is the speaker candidate a_j. In the example of Table 5.1, *Root1* node has two children:

ChildL1: Dilma
ChildR1: que não renuncia

Next, we generate the attribution candidates, permuting the children two by two:

$$\begin{aligned}(q_1, a_1) &= (q_1 = \textit{ChildL1},\ a_1 = \textit{ChildR1}\ ) \\ &= (q_1 = \textit{Dilma},\ a_1 = \textit{que não renuncia}\ ). \\ (q_2, a_2) &= (q_2 = \textit{ChildR1},\ a_2 = \textit{ChildL1}\ ) \\ &= (q_2 = \textit{que não renuncia},\ a_2 = \textit{Dilma}\ ).\end{aligned}$$

The final step is creating two additional pairs of attributions for each quotation candidate. We link them with two *dummy* speakers: *noquote* speaker and *nosubj* speaker. In the example, the set of attributions candidates is:

$$\begin{gathered}(q_1, a_1),\ (q_2, a_2), \\ (q_1, a_3 = \textit{noquote}),\ (q_1, a_4 = \textit{nosubj}), \\ (q_2, a_3 = \textit{noquote}),\ (q_2, a_4 = \textit{nosubj})\end{gathered}$$

Eventually, we input these candidates into our model and, ideally, it selects the right one. What happens if the model selects one of the *dummy* speakers? If the model selects the pair with *noquote* speaker, it is saying that the quotation candidate is not a quotation at all. If the model selects the pair with *nosubj* speaker, it is claiming that the quotation candidate is a quotation, but the speaker is hidden (see section 3.5.1).

PUC-Rio - Certificação Digital Nº 1421603/CA

5.3 Feature Set

Using [6], we generate eleven basic features for each quotation-author combination (q_i, a_j). To exemplify, we use the sentence in Figure 5.1. Each text span is underlined and has an integer subscript that identifies the candidate. As described in section 5.2, the span can be a quotation candidate or an author candidate.

> "Nossa corrupção não é partidária, é decorrente do nosso sistema político"$_1$, afirmou em abril$_2$ o procurador Carlos Fernando dos Santos Lima$_3$.

Figure 5.1: Example to illustrate the generation of the features set

1. *Distance* - contains the number of authors candidates between q_i and a_j. In Figure 5.1, for combination (1,2), we assign *distance0*. For combination (1,3), we assign *distance1*.

2. *Direction* - indicates whether a_j is on the left side or on the right side of q_i. In Figure 5.1, for combination (1,2), we assign *directionRight*. For combination (3, 1), we assign *directionLeft*.

3. *Verb of Speech in Between* - indicates whether there is a verb of speech between q_i and a_j, based on a list of verb of speech. In Figure 5.1, for combination (1,3), we assign *verbOfSpeechInBetween*. For the combination (2,3), we do not assign any tag.

4. *Number of Verbs of Speech in Between* - contains the number of verbs of speech between q_i and a_j. In Figure 5.1, for combination (1,3), we assign *numVerbsOfSpeechInBetween1*. For the combination (2,3), we assign *numVerbsOfSpeechInBetween0*.

5. *Author Candidate in Between* - indicates whether there is a author candidate between q_i and a_j. In Figure 5.1, for combination (1,2), we do not assign any tag. For combination (1,3), we assign *authorCandidateInBetween*.

6. *Author Candidate POS Window* - contains the author candidate POS tag and the POS tags of its nearest ten tokens. In Figure 5.1, for the combinations that contain 2 as an author candidate, we assign the tags *authorPOSWin-5=N*, *authorPOSWin-4=ADJ*, *authorPOSWin-3=QUOTES*, *authorPOSWin-2=COMMA*, *authorPOSWin-1=V*, *authorPOSWin0=PRP*, *authorPOSWin0=N*, *authorPOSWin1=ART*, *authorPOSWin2=N*, *authorPOSWin3=PROP*, *authorPOSWin4=PROP* and *authorPOSWin5=PROP*.

7. *Quotation POS* - contains the quotation POS tag. In Figure 5.1, for the combinations that contain the quotation 2, we assign the tags *QuotationPOS=PRP* and *QuotationPOS=N*.

8. *POS in Between* - contains the POS tags of the tokens between q_i and a_j. In Figure 5.1, for the combination (1,3), we assign the tags *POSInBetween=COMMA*, *POSInBetween=V*, *POSInBetween=PRP* and *POSInBetween=N*.

9. *Bounded Chunk* - indicates whether the quotation candidate is between quotation marks. In Figure 5.1, for the combinations that contain 3 as a quotation candidate, we do not assign any tag.

PUC-Rio - Certificação Digital Nº 1421603/CA

10. *verbOfSpeechNeighborhood* - indicates whether the author candidate is one of the four closest tokens of a verb of speech (two for the left and two for the right). In Figure 5.1, for the combinations that contain 2 as an author candidate, we assign *verbOfSpeechNeighborhood*.

11. *First Letter Upper Case* - indicates whether the first token of the author candidate is a word and its first letter is in upper case. In Figure 5.1, for the combinations that contain 2 as an author candidate, we do not assign any tag.

5.3.1 Binary Features

In general, for NLP problems, we convert the features to binary values. The *binarization* process transforms each feature with m possible values into m binary features, with only one active.

For instance, the feature *Distance*, $\phi_1(x, y)$, contains the number of authors candidates between the quote candidate x and the author candidate y. Suppose that, after processing the dataset, we discover that there are five different values for it, from 0 to 4. We create five binary features, each representing one of the five possible values. Formally, instead of:

$$\phi_1(x, y) = \#\text{ of authors candidates between } x \text{ and } y = 4$$

We represent it as:

$$\begin{aligned}
\phi_{B1}(x, y) &= 1, \text{if } \textit{Distance} \text{ is } 0 \\
&\quad\ 0, otherwise \\
\phi_{B2}(x, y) &= 1, \text{if } \textit{Distance} \text{ is } 1 \\
&\quad\ 0, otherwise \\
\phi_{B3}(x, y) &= 1, \text{if } \textit{Distance} \text{ is } 2 \\
&\quad\ 0, otherwise \\
\phi_{B4}(x, y) &= 1, \text{if } \textit{Distance} \text{ is } 3 \\
&\quad\ 0, otherwise \\
\phi_{B5}(x, y) &= 1, \text{if } \textit{Distance} \text{ is } 4 \\
&\quad\ 0, otherwise
\end{aligned}$$

We can think of the features described at the beginning of this section as binary flags: if we assign the tag, the feature value is 1. Otherwise, its value

PUC-Rio - Certificação Digital Nº 1421603/CA

is 0.

5.3.2 Feature Factorization

To describe the candidate quote-author association (q_i, a_j), we use an input feature vector:

$$\boldsymbol{\Phi}(i,j) = (\phi_1(i,j), \ldots, \phi_M(i,j))$$

where M is the number of all possible features generated in the corpus preprocessing stage and $\phi_k(i,j) = 1$ if the feature belongs to the association (q_i, a_j).

Therefore, for a given output $\boldsymbol{y}$, the global feature *vector* is defined as

$$\boldsymbol{\Phi}(\boldsymbol{x}, \boldsymbol{y}) = \sum_{i=1,\ldots,N} \boldsymbol{\Phi}(i, y_i).$$

5.4 Input-Output Mapping

Finally, we need to translate the input-output defined above to input-output for the prediction problem. Here, the prediction problem is to find non-overlapping quotes associated to authors whose association weights are maximum. This problem can be reduced to the weighted interval scheduling (WIS), described in section 4.4. In order to generate a WIS instance from a quotation extraction input $\boldsymbol{x}$, we create an weighted interval for each association (q_i, a_j). The segment, or span, for this interval is given by the quote segment (s_i, e_i); and given the current model $\boldsymbol{w}$, the interval weight is

$$s(i,j) = \boldsymbol{w} \cdot \boldsymbol{\Phi}(i,j)$$

Since the association $(q_i, a_1), \ldots, (q_i, a_K)$ have the same span in the WIS instance - which is (s_i, e_i) - the WIS algorithm never selects more than one author for q_i. Additionally, overlapping quotes are never selected together. The weight of a complete solution $\boldsymbol{y}$ is then given by

$$\begin{aligned} s(\boldsymbol{y}) &= \sum_{i=1,\ldots,N} s(i, y_i) \\ &= \sum_{i=1,\ldots,N} \boldsymbol{w} \cdot \boldsymbol{\Phi}(i, y_i) \\ &= \boldsymbol{w} \cdot \boldsymbol{\Phi}(x, y) \end{aligned}$$

PUC-Rio - Certificação Digital Nº 1421603/CA

We use a loss function that counts how many quotes have been associated to incorrect authors, that is $\ell(\boldsymbol{y}, \boldsymbol{y'}) = \sum_{i=1,\ldots,N} \mathbf{1}[y_i \neq y'_i]$.

PUC-Rio - Certificação Digital Nº 1421603/CA

6
Experiments

In this chapter, we present the experimental setup and the observed quality of the Structured Perceptron model.

6.1
Evaluation

Our ML model predicts a pair, quotation and author (q, a). In this way, we are solving three tasks at the same time: quotation identification, author identification and association quotation to an author. We evaluate the quality of our model on quotation identification and identification-association to an author, since these are the most frequent results reported in literature.

We consider that the model correctly identifies a quotation if it selects a pair (q, a) in which q is valid and a is not *nosubj*.

6.1.1
Metrics

We evaluate the created model using three metrics: *Precision (P)*, *Recall (R)* and F-score (F or *F1*). *Precision* is what fraction of the items the classifier flags as being in the class actually are in the class.

$$P = \frac{TP}{TP + FP}$$

where TP = *true positives* and FP = *false positives*.

Recall is what fraction of the things that are in the class are detected by the classifier.

$$R = \frac{TP}{TP + FN}$$

where TP = *true positives* and FN = *false negatives*.

F-score is given by

$$F = \frac{2 \times P \times R}{P + R}$$

PUC-Rio - Certificação Digital Nº 1421603/CA

6.2 Experimental Setup

To create and evaluate our model, we split QuoTrees 1.0 into a training and a test sets. The statistics for each set is given on Table 6.1.

Set	*#Tokens*	*#Sentences*	*#Feeds*	*#Quotations*
Training	367,730	19,313	4,142	1,277
Test	93,236	4,826	1,057	386

Table 6.1: Statistics of training and test sets of QuoTrees 1.0

Structured Perceptron is an online algorithm. It means that the order in which examples are processed is relevant for the result model. Taking this into account, we calibrate the model running a 5fold crossvalidation 5 times through the training set. Empirically, we get to 65 epochs and loss weight of 400.

We use a Macbook Air with an Intel Core i7 of 2GHz and 8GB of RAM. We code the system with Python 3.5, using Anaconda 2.4.0 distribution. To create the model, it takes 40 seconds and to evaluate it, 2 seconds.

PUC-Rio - Certificação Digital Nº 1421603/CA

6.3 Quality Results

As mentioned before, we split the results in quotation identification and identification-association to the author. For conciseness, we call the later just *author association*.

6.3.1 Quotation Identification

In Table 6.2, we present the quality of our model assessed in the test set for the quotation identification subtask.

Direct			Indirect			All		
P	*R*	*F*	*P*	*R*	*F*	*P*	*R*	*F*
.70	.67	.68	.88	.87	.87	.70	.97	.81

Table 6.2: Quotation Identification performance

The performance of our model cannot be directly compared to previous work, since the corresponding corpus was specially created for this work and it is not publicly available. In addition, our model is the first one to tackle indirect quotation in Portuguese and to deal with quotes with *missing subject*, in which the source is left implicit (see section 3.5.1).

We suspect that the *missing subject* introduces a high level of complexity to the task. To confirm that, we build another model without the missing subject sentences. The results are shown in Table 6.3. This time, we included some results of previous works just to have an estimation of our quality.

	Direct			Indirect			All		
	P	*R*	*F1*	*P*	*R*	*F*	*P*	*R*	*F*
This work	.80	.77	.78	.91	.91	.91	.83	.98	.90
Pareti et al. [7]	-	-	-	.69	.53	.60	.82	.75	.78

Table 6.3: Quotation Identification without missing subject performance

Pareti et al. [7] report results for two different corpus in English. We show the results for the model that performs better. Their work do not handle *missing subject*, since that does not happen in English.

6.3.2 Author Association

In Table 6.4, we present the quality of our model assessed in the test set for the association subtask.

Direct			Indirect			All		
P	*R*	*F*	*P*	*R*	*F*	*P*	*R*	*F*
.70	.67	.68	.88	.87	.87	.54	.79	.66

Table 6.4: Quotation Identification and Author Association performance

In addition, we measure the results for the same task in the model described in 6.3.1, that do not handle *missing subject* quotations. Again, we confirm an improvement in the results, as shown in Table 6.5.

	Direct			Indirect			All		
	P	*R*	*F*	*P*	*R*	*F*	*P*	*R*	*F*
This work	.80	.77	.78	.91	.91	.91	.72	.86	.78
Fernandes [6]	.83	.71	.77	-	-	-	-	-	-

Table 6.5: Quotation Identification and Author Association without missing subject performance

Fernandes [6] uses a different corpus. He just presents results for the complete task (identification and association) and does not tackle indirect quotations, although he tackles mixed quotations. He does not deal with *missing subject*.

PUC-Rio - Certificação Digital Nº 1421603/CA

6.4 Error Analysis

We analyse two categories of errors: association and identification errors. The former are those in which the model associates a wrong author for a correctly identified quote. The identification errors are those in which the model classifies a text span as quotation when it is not. For association errors, we have noticed that most of them is misclassified *missing subject* quotes, as we confirm in the section above. However, in the majority of the cases in which the model "wrongly" identifies a quotation, the text span is actually a real quotation, but it is not annotated in the corpus. This is due to annotation mistakes or, rarely, a type of quotation that does not fit in any of the rules. Most of the annotation mistakes are propagated by the use of Floresta, which is only source of QuoTrees 1.0 that is not revised manually. These results confirm the discussion in section 3.4.1, that the ML are flexible enough to correctly deal with unseen examples.

PUC-Rio - Certificação Digital Nº 1421603/CA

7
Conclusions

Quotation Extraction consists of identifying quotations from a text and associating them to their authors. We propose a Quotation Extraction system for the Portuguese language.

Few are the works in Quotation Extraction that address indirect quotations, that, in many cases, are the majority type presented in a text. Our proposal is in compliance with the latest researches on the subject, since we tackle direct and indirect quotation using a Machine Learning approach. Machine Learning models usually present stronger generalization power compared to human-derived ones due to their capacity to adapt to different writing styles. In human-derived models, even small changes in the writing style may need several modifications in the human-derived rule set. In addition, we are able to easily adapt our model into other languages, needing just a list of verbs of speech for a given language.

We create a Structured Perceptron model based on the weight interval scheduling problem (SP-WIS). In this model, we find a maximum-weight subset of non-overlapping tasks, where each task represents a combination of quotation and author candidate.

In this work, we also built QuoTrees 1.0, the first corpus in Portuguese annotated with direct and indirect quotations and associations between quotations and authors. In addition, we annotate quotes with missing subject and analyse some particularities of Portuguese language that create more complexity to the Quotation Extraction task.

Our SP-WIS model obtains a new state-of-the-art F-score of 66% for QuoTrees 1.0, tackling direct and indirect quotation, in addition to *missing subject* quotes.

We also build a model without the missing subject in order to compare the performance of our work against previous works. This model obtains a F-score of 78%.

7.1

PUC-Rio - Certificação Digital Nº 1421603/CA

Future work

Future work will include extending the model to extract all kind of attributions and not just quotations (i.e. beliefs, eventualities, and facts). In order to do this, we need to enhance the annotation of QuoTrees 1.0. Moreover, we will tackle cases where the cues are not just verbs of speech and the particular case of nested quotations. We also could modify the model to solve the task without relying on a list of verbs of speech. Finally, we could apply our system to other language.

PUC-Rio - Certificação Digital Nº 1421603/CA

Bibliography

[1] JOHNSON, M.. **How the statistical revolution changes (computational) linguistics**. In: PROCEEDINGS OF THE EACL 2009 WORKSHOP ON THE INTERACTION BETWEEN LINGUISTICS AND COMPUTATIONAL LINGUISTICS: VIRTUOUS, VICIOUS OR VACUOUS?, p. 3–11. Association for Computational Linguistics, 2009.

[2] TSOCHANTARIDIS, I.; HOFMANN, T.; JOACHIMS, T. ; ALTUN, Y.. **Support vector machine learning for interdependent and structured output spaces**. In: PROCEEDINGS OF THE TWENTY-FIRST INTERNATIONAL CONFERENCE ON MACHINE LEARNING, ICML '04, p. 104–, New York, NY, USA, 2004. ACM.

[3] ALTUN, Y.; TSOCHANTARIDIS, I.; HOFMANN, T. ; OTHERS. **Hidden markov support vector machines**. In: ICML, volumen 3, p. 3–10, 2003.

[4] COLLINS, M.. **Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms**. In: PROCEEDINGS OF THE ACL-02 CONFERENCE ON EMPIRICAL METHODS IN NATURAL LANGUAGE PROCESSING-VOLUME 10, p. 1–8. Association for Computational Linguistics, 2002.

[5] LIANG, P.; BOUCHARD-CÔTÉ, A.; KLEIN, D. ; TASKAR, B.. **An end-to-end discriminative approach to machine translation**. In: PROCEEDINGS OF THE 21ST INTERNATIONAL CONFERENCE ON COMPUTATIONAL LINGUISTICS AND THE 44TH ANNUAL MEETING OF THE ASSOCIATION FOR COMPUTATIONAL LINGUISTICS, p. 761–768. Association for Computational Linguistics, 2006.

[6] FERNANDES, W. P. D.. **Quotation extraction for portuguese**. Master's thesis, Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro, 2012.

[7] PARETI, S.; O'KEEFE, T.; KONSTAS, I.; CURRAN, J. R. ; KOPRINSKA, I.. **Automatically detecting and attributing indirect quotations.** In: EMNLP, p. 989–999, 2013.

PUC-Rio - Certificação Digital Nº 1421603/CA

[8] QUINTÃO, M. E.. **Quotation attribution for portuguese news corpora**. Master's thesis, Técnico Lisboa/UTL, Portugal, April 2014.

[9] PARETI, S.. **Towards a discourse resource for italian: Developing an annotation schema for attribution**. Master's thesis, Università degli Studi di Pavia, Pavia, 29 September 2009.

[10] PARETI, S.. **A database of attribution relations.** In: LREC, p. 3213–3217. Citeseer, 2012.

[11] DAVIDSON, D.. **Quotation**. Theory and decision, 11(1):27–40, 1979.

[12] STERNBERG, M.. **Proteus in quotation-land: Mimesis and the forms of reported discourse**. Poetics today, 3(2):107–156, 1982.

[13] CAPPELEN, H.; LEPORE, E.. **Varieties of quotation**. Mind, 106(423):429–450, 1997.

[14] RECANATI, F.. **Open quotation**. Mind, 110(439):637–687, 2001.

[15] PARETI, S.. **Annotating attribution relations and their features**. In: PROCEEDINGS OF THE FOURTH WORKSHOP ON EXPLOITING SEMANTIC ANNOTATIONS IN INFORMATION RETRIEVAL, p. 19–20. ACM, 2011.

[16] PARETI, S.. **Attribution: a computational approach**. Doctoral thesis, The University of Edinburgh, 2015.

[17] BERGLER, S.; DOANDES, M.; GERARD, C. ; WITTE, R.. **Attributions**. Exploring Attitude and Affect in Text: Theories and Applications, Technical Report SS-04-07, p. 16–19, 2004.

[18] FREITAS, C.; FREITAS, B. ; SANTOS, D.. **Quemdisse? reported speech in portuguese**. In: Chair), N. C. C.; Choukri, K.; Declerck, T.; Goggi, S.; Grobelnik, M.; Maegaard, B.; Mariani, J.; Mazo, H.; Moreno, A.; Odijk, J. ; Piperidis, S., editors, PROCEEDINGS OF THE TENTH INTERNATIONAL CONFERENCE ON LANGUAGE RESOURCES AND EVALUATION (LREC 2016), Paris, France, may 2016. European Language Resources Association (ELRA).

[19] HARRY, J. C.. **Journalistic quotation: Reported speech in newspapers from a semiotic-linguistic perspective**. Journalism, 15(8):1041–1058, 2014.

PUC-Rio - Certificação Digital Nº 1421603/CA

[20] POULIQUEN, B.; STEINBERGER, R. ; BEST, C.. **Automatic detection of quotations in multilingual news**. In: PROCEEDINGS OF RECENT ADVANCES IN NATURAL LANGUAGE PROCESSING, p. 487–492, 2007.

[21] KRESTEL, R.; BERGLER, S.; WITTE, R. ; OTHERS. **Minding the source: Automatic tagging of reported speech in newspaper articles**. Reporter, 1(5):4, 2008.

[22] SCHNEIDER, N.; HWA, R.; GIANFORTONI, P.; DAS, D.; HEILMAN, M.; BLACK, A.; CRABBE, F. L. ; SMITH, N. A.. **Visualizing topical quotations over time to understand news discourse**. 2010.

[23] DE LA CLERGERIE, É.; SAGOT, B.; STERN, R.; DENIS, P.; RECOURCÉ, G. ; MIGNOT, V.. **Extracting and visualizing quotations from news wires**. In: LANGUAGE AND TECHNOLOGY CONFERENCE, p. 522–532. Springer, 2009.

[24] DE MORAIS, L. A. D. F.; NUNES, S. S. ; OTHERS. **Automatic extraction of quotes and topics from news feeds**. In: DSIE09-4TH DOCTORAL SYMPOSIUM ON INFORMATICS ENGINEERING, 2009.

[25] FREITAS, C.; ROCHA, P. ; BICK, E.. **Floresta sintá (c) tica: bigger, thicker and easier**. In: INTERNATIONAL CONFERENCE ON COMPUTATIONAL PROCESSING OF THE PORTUGUESE LANGUAGE, p. 216–219. Springer, 2008.

[26] BICK, E.. **The Parsing System" Palavras": Automatic Grammatical Analysis of Portuguese in a Constraint Grammar Framework**. Aarhus Universitetsforlag, 2000.

[27] BICK, E.; SANTOS, D.. **Projecto floresta sintá(c)tica**. `http://www.linguateca.pt/Floresta/`. Accessed: 2016-12-18.

[28] FREITAS, C.; AFONSO, S.. **Bíblia Florestal: Um manual lingüístico da floresta sintá(c)tica.** `http://www.linguateca.pt/floresta/BibliaFlorestal/completa.html`. Accessed: 2016-11-14.

[29] FREITAS, B.. **O dizer em português: diálogos entre tradução, descrição e linguística computacional**. Master's thesis, Pontifícia Universidade Católica do Rio de Janeiro, April 2015.

[30] CUEVAS, R. R. M.; PARABONI, I.. **A machine learning approach to portuguese pronoun resolution**. In: IBERO-AMERICAN CONFERENCE ON ARTIFICIAL INTELLIGENCE, p. 262–271. Springer, 2008.

PUC-Rio - Certificação Digital Nº 1421603/CA

[31] DE SOUZA, J. G. C.; GONÇALVES, P. N. ; VIEIRA, R.. **Learning coreference resolution for portuguese texts**. In: INTERNATIONAL CONFERENCE ON COMPUTATIONAL PROCESSING OF THE PORTUGUESE LANGUAGE, p. 153–162. Springer, 2008.

[32] KOO, T.; RUSH, A. M.; COLLINS, M.; JAAKKOLA, T. ; SONTAG, D.. **Dual decomposition for parsing with non-projective head automata**. In: PROCEEDINGS OF THE 2010 CONFERENCE ON EMPIRICAL METHODS IN NATURAL LANGUAGE PROCESSING, p. 1288–1298. Association for Computational Linguistics, 2010.

[33] DOS SANTOS, C. N.; MILIDIÚ, R. L. ; RENTERÍA, R. P.. **Portuguese part-of-speech tagging using entropy guided transformation learning**. In: INTERNATIONAL CONFERENCE ON COMPUTATIONAL PROCESSING OF THE PORTUGUESE LANGUAGE, p. 143–152. Springer, 2008.

[34] ROSENBLATT, F.. **The perceptron: a probabilistic model for information storage and organization in the brain.** Psychological review, 65(6):386, 1958.

[35] CHOI, J. Y.; CHOI, C.-H.. **Sensitivity analysis of multilayer perceptron with differentiable activation functions**. IEEE Transactions on Neural Networks, 3(1):101–107, 1992.

[36] KARLIK, B.; OLGAC, A. V.. **Performance analysis of various activation functions in generalized mlp architectures of neural networks**. International Journal of Artificial Intelligence and Expert Systems, 1(4):111–122, 2011.

[37] CRAMMER, K.; SINGER, Y.. **Ultraconservative online algorithms for multiclass problems**. Journal of Machine Learning Research, 3(Jan):951–991, 2003.

[38] ROLLER, B. T. C. G. D.. **Max-margin markov networks**. Advances in neural information processing systems, 16:25, 2004.

PUC-Rio - Certificação Digital Nº 1421603/CA

A
List of speech verbs by rule

Table A.1: List of speech verbs for Rule 1.

Speech Verbs: Rule 1				
acentuar	assinalar	contabilizar	divertirse	gracejar
aconselhar	assumir	contar	dizer	gritar
acreditar	atacar	contareu	dizereu	idealizar
acrescentar	atestar	contarnós	dizernós	ilustrar
acusar	atirar	contestar	elogiar	imaginar
adiantar	avaliar	continuar	emendar	indagar
admitir	avisar	contrapor	empolgarse	indicar
advertir	berrar	convidar	enfatizar	indignarse
aferir	bradar	cortar	ensinar	informar
afirmar	brincar	criticar	entregar	insistir
afirmarse	cantar	decidir	entusiasmarse	interrogar
alegar	categorizar	declarar	enumerar	interromper
alegrarse	chamar	decretar	esclarecer	inventar
alertar	citar	defender	esperar	ironizar
alfinetar	clarificar	defenderse	estranhar	jurar
ameaçar	colocar	definir	exclamar	justificar
amenizar	comemorar	demonstrar	exemplificar	justificarse
analisar	comentar	denunciar	explicar	lamentar
animarse	comentarse	desabafar	explicarse	lamentarse
anotar	começar	desafiar	expor	lembrar
anunciar	comparar	despistar	exultar	minimizar
apelar	complementar	destacar	falar	mostrarse
apoiar	completar	detalhar	festejar	murmurar
apontar	concluir	detonar	filosofar	narrar
apostar	concordar	devolver	finalizar	notar
argumentar	confessar	diferenciar	frisar	notar
arrematar	confirmar	discorrer	gabarse	observar
arriscar	considerar	discursar	garantir	opinar
assegurar	constatar	disparar	gargalhar	orgulharse

PUC-Rio - Certificação Digital Nº 1421603/CA

Speech Verbs: Rule 1 - continued from previous page				
pedir	prosseguir	referirse	responderele	sorrir
pensar	protestar	reflectir	ressaltar	sublinhar
perguntar	provocar	refletir	resumir	sublinharse
planejar	queixarse	reforçar	retomar	sugerir
ponderar	questionar	registrar	retorquir	sustentar
pontuar	questionarse	regozijarse	retratar	temer
precisar	reafirmar	reiterar	revelar	teorizar
preconizar	reagir	relatar	rir	testemunhar
prever	rebater	relembrar	rirse	torcer
proclamar	recear	rematar	salientar	vaticinar
profetizar	reclamar	rememorar	sentenciar	
prometer	reconhecer	repetir	simplificar	
propor	recordar	resignarse	sintetizar	
prosear	referir	responder	situar	
End of Table				

PUC-Rio - Certificação Digital Nº 1421603/CA

Table A.2: List of speech verbs for Rule 2.

Speech Verbs: Rule 2				
aconselhar	brincar	desabafar	insistir	propor
acrescentar	cantar	destacar	interrogar	prosseguir
acusar	chamar	discordar	interrogarse	provocar
adiantar	chorar	disparar	interromper	questionar
advertir	clamar	dizer	intimar	rebater
afirmar	comentar	emendar	ironizar	reclamar
ajudar	complementar	enfatizar	justificar	recomendar
alertar	completar	ensinar	lamentar	recordar
alfinetar	concluir	esclarecer	lembrar	refletir
anunciar	concordar	exclamar	narrar	registrar
apontar	confessar	exemplificar	notar	relatar
argumentar	confidenciar	explicar	observar	relembrar
arrematar	confirmar	falar	pedir	rematar
arriscar	contar	filosofar	pensar	repetir
assegurar	continuar	finalizar	perguntar	replicar
assinalar	convidar	garantir	precisar	resmungar
atacar	declarar	gritar	proclamar	responder
atestar	defender	hesitar	profetizar	ressaltar
avisar	definir	indagar	prometer	resumir

Speech Verbs: Rule 2 - continued from previous page				
retrucar	ripostar	soltar	sugerir	
revelar	sentenciar	sorrir	teorizar	
End of Table				

Table A.3: List of speech verbs for Rule 3.

Speech Verbs: Rule 3				
acentuar	cantar	ditar	mencionar	registrar
aconselhar	chamar	divulgar	negar	rejeitar
acrescentar	chutar	dizer	noticiar	relatar
acrescer	citar	enfatizar	observar	relembrar
adiantar	colocar	esclarecer	opinar	repetir
admitir	combinar	especificar	ordenar	replicar
advertir	comentar	especular	pedir	resmungar
advogar	completar	estimar	perguntar	responder
afiançar	comunicar	estipular	ponderar	ressaltar
afirmar	conceder	exclamar	precisar	revelar
alardear	concluir	exemplificar	preferir	rezar
alegar	concordar	exigir	pregar	salientar
alertar	confessar	explicar	presumir	sentenciar
aludir	confiar	explicitar	prevenir	sinalizar
analisar	confidenciar	falar	prever	solicitar
antever	confirmar	frisar	proclamar	sublinhar
anunciar	constatar	garantir	proibir	sugerir
apontar	contar	gritar	prometer	suplicar
apostar	contestar	impor	pronunciar	suspeitar
apregoar	criticar	incentivar	propor|propôr	sustentar
apurar	declarar	indagar	queixar	teimar
argumentar	definir	informar	questionar	testemunhar
assegurar	deliberar	insinuar	reafirmar	vaticinar
asseverar	denunciar	insistir	realçar	
assinalar	desconfiar	inventar	reclamar	
assumir	desmentir	julgar	recomendar	
atestar	destacar	jurar	reconhecer	
avaliar	detalhar	justificar	recordar	
avisar	determinar	lamentar	recusar	
berrar	discordar	lembrar	referir	
brincar	discutir	manter	reforçar	

PUC-Rio - Certificação Digital Nº 1421603/CA

Table A.4: List of speech verbs for Rule 4.

Speech Verbs: Rule 4				
acentuar	assinalar	contabilizar	divertirse	gracejar
aconselhar	assumir	contar	dizer	gritar
acreditar	atacar	contareu	dizereu	idealizar
acrescentar	atestar	contarnós	dizernós	ilustrar
acusar	atirar	contestar	elogiar	imaginar
adiantar	avaliar	continuar	emendar	indagar
admitir	avisar	contrapor	empolgarse	indicar
advertir	berrar	convidar	enfatizar	indignarse
aferir	bradar	cortar	ensinar	informar
afirmar	brincar	criticar	entregar	insistir
afirmarse	cantar	decidir	entusiasmarse	interrogar
alegar	categorizar	declarar	enumerar	interromper
alegrarse	chamar	decretar	esclarecer	inventar
alertar	citar	defender	esperar	ironizar
alfinetar	clarificar	defenderse	estranhar	jurar
ameaçar	colocar	definir	exclamar	justificar
amenizar	comemorar	demonstrar	exemplificar	justificarse
analisar	comentar	denunciar	explicar	lamentar
animarse	comentarse	desabafar	explicarse	lamentarse
anotar	começar	desafiar	expor	lembrar
anunciar	comparar	despistar	exultar	minimizar
apelar	complementar	destacar	falar	mostrarse
apoiar	completar	detalhar	festejar	murmurar
apontar	concluir	detonar	filosofar	narrar
apostar	concordar	devolver	finalizar	notar
argumentar	confessar	diferenciar	frisar	notar
arrematar	confirmar	discorrer	gabarse	observar
arriscar	considerar	discursar	garantir	opinar
assegurar	constatar	disparar	gargalhar	orgulharse
pedir	prosseguir	referirse	responderele	sorrir
pensar	protestar	reflectir	ressaltar	sublinhar
perguntar	provocar	refletir	resumir	sublinharse
planejar	queixarse	reforçar	retomar	sugerir
ponderar	questionar	registrar	retorquir	sustentar
pontuar	questionarse	regozijarse	retratar	temer
precisar	reafirmar	reiterar	revelar	teorizar

PUC-Rio - Certificação Digital Nº 1421603/CA

Speech Verbs: Rule 4 - continued from previous page				
preconizar	reagir	relatar	rir	testemunhar
prever	rebater	relembrar	rirse	torcer
proclamar	recear	rematar	salientar	vaticinar
profetizar	reclamar	rememorar	sentenciar	
prometer	reconhecer	repetir	simplificar	
propor	recordar	resignarse	sintetizar	
prosear	referir	responder	situar	
End of Table				

Table A.5: List of speech verbs for Rule 5.

Speech Verbs: Rule 5				
acentuar	confirmar	evocar	observar	responder
acrescentar	considerar	exigir	pedir	ressaltar
adiantar	convidar	explicar	perguntar	resumir
admitir	convocar	expor	prometer	revelar
afirmar	crer	expressar	propor	rezar
agradecer	criticar	exprimir	prosseguir	rir
ameaçar	cumprimentar	frisar	protestar	salientar
analisar	debater	gritar	provocar	solicitar
anunciar	decidir	homenagear	querer saber	sorrir
apontar	declarar	idealizar	questionar	sublinhar
argumentar	defender	impor	realçar	sugerir
avaliar	desafiar	informar	recitar	supor
brincar	desculpar	insistir	reclamar	sustentar
cantar	desmentir	inventar	reconhecer	xingar
chamar	destacar	julgar	referir	
citar	discutir	jurar	refletir	
comemorar	divulgar	lamentar	registrar	
comentar	dizer	mentir	rejeitar	
concluir	elogiar	narrar	relatar	
concordar	enfatizar	negar	relembrar	
confessar	estipular	notar	repetir	
End of Table				

PUC-Rio - Certificação Digital Nº 1421603/CA