UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE CIÊNCIA DA COMPUTAÇÃO

FELIPE BUENO DA ROSA

Study of a Deep Learning Approach to Named Entity Recognition for Portuguese

Work presented in partial fulfillment
of the requirements for the degree of
Bachelor in Computer Science

Advisor: Prof. Dr. Dante Barone
Coadvisor: Eduardo Cortes

Porto Alegre
December 2018

ABSTRACT

This work aims to advance the state-of-the-art in named entity recognition for the Portuguese language using deep learning. It proposes the addition of part-of-speech tagging to a system composed of a bidirectional LSTM and a CNN neural network architecture. It evaluates the new system under the guidelines of the HAREM contest and compares its results to those of other participants.

Keywords: Named entity recognition. deep learning. HAREM. natural language processing.

Estudo de uma Abordagem de Aprendizagem Profunda para o Reconhecimento de Entidades Nomeadas para o Português

RESUMO

Este trabalho almeja avançar o estado-da-arte em reconhecimento de entidades mencionadas para a língua portuguesa usando aprendizagem profunda. Ele propõe a adição de etiquetagem de classes gramaticais a um sistema composto das arquiteturas de redes neurais LSTM bidirecional e CNN. Ele avalia o novo sistema sob as diretrizes da competição HAREM e compara os seus resultados com aqueles dos outros participantes.

Palavras-chave: Reconhecimento de entidades mencionadas. aprendizagem profunda. HAREM. processamento de linguagem natural..

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

HAREM	Harem é uma Avaliação de Reconhecedores de Entidades Mencionadas
CoNLL	Conference on Computational Natural Language Learning
MET	Multiligual Entity Task
ACE	Automatic Content Extraction
REMBRANDT	Reconhecimento de Entidades Mencionadas Baseado em Relações e Análise Detalhada do Texto
XIP	Xerox Incremental Parser
LSTM	Long-Short Term Memory
CNN	Convolutional Neural Network
POS	Part of Speech
SAHARA	Serviço de Avaliação HAREM Automático

CONTENTS

1 INTRODUCTION

Named entities are textual expressions within a document written in natural language that refer to unique real-world objects. These expressions manifest themselves through proper names, such as the names of persons (e.g *John Hime*), names of organizations (e.g. *Boston Chicken Corp.*), and names of locations (e.g. *Northern California*). In many languages there is a convention of writing these expressions with capitalized words, although some of the words within the expression may not be capitalized (e.g. *Graduate School of Business*) and some capitalized words may not be named entities (e.g. any sentence-starting word is capitalized in English). Furthermore, there is also another group of expressions that are considered named entities despite not being proper names: temporal expressions (e.g. *January 1990*) and number expressions (e.g. *$42.1 million*). That encloses the classical definition of named entity (CHINCHOR; ROBINSON, 1998), but variations of this model can be found across different authors. (SANG; MEULDER, 2003; DODDINGTON et al., 2004; SANTOS et al., 2008)

Recognizing a named entity involves two subtasks: identification and classification (SANTOS; CARDOSO, 2007). In the identification subtask, the expression corresponding to a named entity needs to be isolated from the rest of the surrounding text. In the classification subtask, the semantic category of the named entity must be found out—one has to classify it into, say, a name expression, a temporal expression, or a number expression. The task of named entity recognition finds its purpose in helping to give semantic structure to previously unstructured documents, an important contribution to larger tasks, such as those of information retrieval, question answering, and automatic summarization.

Two main approaches have been employed to solve named entity recognition (AMARAL et al., 2013). One of them characterizes itself by the use of hand-crafted rules for figuring out the presence and type of named entities. These resources come in the form of lists of words and sentence structure templates that are manually produced by the work of human specialists. In general, this approach is the one that provides the best results since the system is carefully designed to take advantage of the particularities of the target language. On the other hand, the disadvantage is that such systems are restricted to a single linguistic domain and will perform poorly if transferred to another. Moreover, the work done by the human specialists makes this approach rather expensive. The other approach relies on using machine learning to find the best rules for identifying and classifying named entities based on statistical inferences made automatically over previously annotated data. This approach still requires human intervention to annotate the training data, but the effort needs to be done only once for each language and it can be reused by any system. Therefore, one of the advantages of this approach is that it is cheap. The other advantage is that the systems made in this way are less dependent on the target language. Conversely, the disadvantage is that the performance of such systems often ends up being worse than the performance reached by the best systems that employed hand-crafted rules. Beyond those two options, many systems combine the two approaches to create a hybrid one, capturing the advantages of machine learning and hand-crafted rules while attempting to get rid of the disadvantages of each one of them. Different systems can draw more or less from each approach when designing their own.

For the Portuguese language, one of them main promoters of research in named entity recognition was Linguateca through its event HAREM. The event was a shared evaluation contest that happened in two editions plus a special one from 2005 to 2008 (SANTOS; CARDOSO, 2007). HAREM not only defined the state-of-the-art for Portuguese named entity recognition, but it set a new standard of guidelines for evaluating the quality of the systems, in many ways different from its predecessors. The three best participants of the second edition of HAREM all used hand-crafted rules, with the best participant, Priberam, achieving an F-score of 57.11% on the classical track (AMARAL et al., 2008; CARDOSO, 2008a; HAGèGE; BAPTISTA; MAMEDE, 2008). When this result is compared to that of a deep learning based system that achieved an F-score of 91.62% on CoNLL-2003 (CHIU; NICHOLS, 2015), it gives the impression that there is still room for overcoming Priberam's performance.

The system that Chiu and Nichols (2015) proposed is a bidirectional LSTM-CNN neural network built primarily for the English language that makes minimal use of external resources. The bidirectional LSTM (Long Short Term Memory) layer is responsible to taking in account the context of each word in a sentence and the CNN (Convolutional Neural Network) layer extracts character-level features. Due to the language-independent nature of machine learning systems, we believe that the system of Chiu and Nichols (2015) would perform well in a different environment. This work aims to answer the question of whether adapting this system to the HAREM requirements and running it against the other participants on the classical track will make it surpass the F-score of its competitors. Since named entities are generally noun phrases, we also want to test whether equipping the system of Chiu and Nichols (2015) with part-of-speech tagging will make its F-score higher. If our hypothesis that this system will surpass HAREM's best participants turns out to be true, then that will represent an advancement for the state-of-the-art in Portuguese named entity recognition.

This work is structured in the following way: in Chapter 2 we present the research context of named entity recognition and go through the workings of the HAREM contest; in Chapter 3 we detail the strategies used by other systems that had the same aim as us, in particular, we analyze the best systems that participated in the HAREM contest; in Chapter 4 we explain the technologies used to build our system and the methodology used to evaluate it; in Chapter 5 we show the performance achieved by our system and how it relates to its competitors; in Chapter **??** we hypothesize the factors that influenced the performance of the system and suggest areas to be researched in future works.

2 BACKGROUND

In this chapter we present the contest used for evaluating our proposal, its history, and its guidelines. We start by talking about the predecessors of HAREM and how each of them contributed to the research area. Then we talk about HAREM in more detail, its distinctive characteristics, and we give an overview of its rules. At the end we walk through the identification and classification guidelines of the contest, listing each of the categories, types, and subtypes available.

2.1 Named Entity Recognition Contests

HAREM was a shared evaluation contest in the area of named entity recognition for Portuguese. A shared evaluation contest is a model of publishing where several approaches to solve the same problem are compared in a fair and impartial way with the goal of stimulating progress in the area. Before shared evaluation contests, each published system was evaluated exclusively by its own author, making it hard to compare one system to another. These contests came as a way to standardize the evaluation method for all these systems. The sixth edition of MUC (SUNDHEIM, 1995) was the first shared evaluation contest to propose that the task of named entity recognition should be done independently. Up to that point, this task was measured in conjunction with the broader task of retrieving information from text. In MUC's definition of named entity, each entity would belong to a type, and each type would have a certain number of subtypes. The types of named entities were:

- *ENAMEX:* for entity name expressions, with the subtypes *NAME*, *ORGANIZATION*, and *LOCATION*.
- *TIMEX:* for temporal expressions, with the subtypes *DATE* and *TIME*.
- *NUMEX:* for number expressions, with the subtypes *MONEY* and *PERCENT*.

MUC achieved very good results in the task and the best participant reached an F-score comparable to those of humans, 96.42%. Although the task seemed easy, MUC's results were valid only for English and other languages have been left unexplored. The task of exploring named entity recognition for languages other than English was taken by subsequent contests. MET (MERCHANT; OKUROWSKI; CHINCHOR, 1996) was based on the same guidelines laid out by MUC, but the languages targeted were Spanish, Japanese, Chinese, and French. CoNLL (SANG; MEULDER, 2003) aimed to investigate language-independent solutions for named entity recognition by relying on Dutch, English, German, and Spanish datasets. Its guidelines were a bit different from the ones used by MUC, as the only categories it had were *LOC* (for locations), *PER* (for persons), *ORG* (for organizations), and the catch-all *MISC* (for miscellaneous). ACE (DODDINGTON et al., 2004) went further on the task by proposing the track *EDT - Entity Detection and Tracking* in its contest. In this track, the entities should not only be *detected*, i.e, identified, but they also should be *tracked*. In other words, any mention made to an entity should be taken in account, no matter if it happened in the form of a proper name, a pronoun, or a description. The category hierarchy employed by ACE was also larger than the one defined by MUC. Besides the classical categories of Person, Organization, and Location, there were also the categories of Facility, Weapon, Vehicle, and Geo-Political Entity, each one of them further divided into several subtypes. Finally, HAREM was concerned in solving named entity recognition for Portuguese and initially took inspiration from MUC, but soon ended up developing its own distinct characteristics. In the next section we discuss the HAREM contest in more detail.

2.2 HAREM

The idea of creating a shared evaluation contest for named entity recognition in Portuguese was motivated by disagreements over what the concept of named entity should entail. The designers of HAREM were not satisfied with the scope of categories available in MUC and wanted it to be as broad and fine-grained as possible. They also wanted the context surrounding the named entities to play a greater role when deciding its category. HAREM's documentation even translates the expression "*named entity*" to "*entidade mencionada*" (mentioned entity) to highlight the dependence of the entities on context. The first edition of HAREM happened in 2004, followed by a special edition known as Mini-HAREM in 2006 and then the second edition in 2007. The Mini-HAREM was a second evaluation with the same participants as the first edition, it happened because many of the systems in that edition were delivered after the deadline due to misunderstandings about the rules of the contest. The classification system of the second edition of HAREM had the categories *PESSOA*, *ORGANIZACAO*, *LOCAL*, *OBRA*, *VALOR*, *TEMPO*, *ACONTECIMENTO*, *COISA*, *ABSTRACCAO*, and *OUTRO*, each one of them further divided into several types and some of the types even further divided into subtypes (CARVALHO et al., 2008). Despite the nomenclature of HAREM being given in the European dialect, the contest was aimed at all dialects of Portuguese.

The annotation scheme used in the second edition of HAREM is based on the XML syntax (SANTOS et al., 2008). The text of each named entity is enclosed by EM tags containing a list of attributes detailing the class of the entity, namely, *ID*, *CATEG*, *TIPO*, and *SUBTIPO*. The attribute *ID* is the only one that is mandatory and should contain a unique string identifying the named entity. The attributes *CATEG*, *TIPO*, and *SUBTIPO* correspond, respectively, to the category, type, and subtype of the named entity. This scheme does not allow one named entity to be inside of another. An example of annotation is:

1. os <EM ID="1" CATEG="PESSOA" TIPO="GRUPOMEMBRO">**Beatles**</EM>.

The most distinctive characteristics of HAREM in contrast to other contests is the attention given to metonymy and vagueness. Metonymy is the idea that the classification of a named entity depends entirely of the context of such entity, and not of certain intrinsic qualities of it. For example, the expression *Portugal* seems to necessarily belong to the category *LOCAL*, but depending on its context it can instead belong to several others:

1. Regressou então a <EM ID="ub-67792-10" CATEG="LOCAL" TIPO="HUMANO" SUBTIPO= "PAIS">**Portugal**</EM>, onde iniciou meteórica carreira na experimentação de novas formas de expressão.
2. O acordo político quanto à revisão foi obtido durante a Presidência Alemã, tendo cabido a <EM ID="a46996-5" CATEG="ORGANIZACAO" TIPO="ADMINISTRACAO">**Portugal**</EM> concluir o processo de revisão.
3. Este debate passou completamente ao lado de <EM ID="2-dftre765-" CATEG="PESSOA" TIPO="POVO">**Portugal**</EM>
4. O problema do PSD é começar a ter só um <EM ID="ub-24360-32" CATEG="ABSTRACCAO" TIPO="IDEIA">**Portugal**</EM> ou dois dentro de si

Therefore, if *Portugal* refers to the physical area within the geopolitical borders of the same named country, then it belongs to *LOCAL*. On the other hand, if it refers to the governing entity that rules said area, then it belongs to *ORGANIZACAO*. However, if it refers to the people inhabiting the country as a group, then it belongs to *PESSOA*. Now, if *Portugal* refers to the representation of an abstract idea, then it belongs to *ABSTRACCAO*. There are many other meanings that could be attributed to this named entity, HAREM expects its participants to choose the right one based exclusively on how the expression relates to the rest of the text.

The other distinctive characteristic of HAREM is the possibility to allow vagueness both in the identification and in the classification of named entities. That means that the same expression can be broken down into alternative, overlapping named entities and that a named entity can belong to more than one category, type or subtype. An example of identification vagueness happens in the sentence *"aproximava a Igreja de Inglaterra do calvinismo"*, which can be annotated as:

1. aproximava a <ALT> <EM ID="2-dftre765-10" CATEG="ABSTRACCAO" TIPO="DISCIPLINA"> **Igreja de Inglaterra**</EM> | <EM ID="2-dftre765-106-a" CATEG="ABSTRACCAO" TIPO= "DISCIPLINA">**Igreja**</EM> de <EM ID="2-dftre765-1" CATEG="LOCAL" TIPO="HUMANO" SUBTIPO="PAIS">**Inglaterra**</EM> </ALT> do calvinismo.

The convention for denoting identification vagueness in the annotation scheme is to enclose all the alternative identifications within an *ALT* tag, with each identification separated by a | sign. In this way, *Igreja de Inglaterra* can be thought of both as a single named entity indicating a discipline and as two named entities, the non-country-specific discipline *Igreja* and the country *Inglaterra*, separated by the word *de*. An example of the other kind of vagueness, classification vagueness, happens in the following sentence:

1. Pela mão do ministro Freitas do Amaral, e sem necessidade alguma, <EM ID= "a66435-10" CATEG="ORGANIZACAO|PESSOA" TIPO="ADMINISTRACAO|POVO">**Portugal**</EM> foi enxovalhado, coberto de vergonha e de cobardia, por um dos mais tristes textos políticos que alguém já escreveu.

The alternative classifications are expressed in the annotation scheme by giving to the attributes *CATEG*, *TIPO*, and *SUBTIPO* multiple values separated by | signs. The values must be listed in the correct order across the attributes, in such way as making the n-th value of *CATEG* be a container class of the n-th value of *TIPO* and so on. In the example given, *Portugal* can refer either to the government of the country or to the population that inhabits there.

2.3 Identification Guidelines

HAREM determines that every named entity should contain at least one uppercase letter or one digit, except for *TEMPO* entities (SANTOS et al., 2008). Named entities can also span several contiguous words, even with some—but not all—of these words being completely in lowercase. Therefore, *médio oriente* is not a named entity, while *ministro da Administração Interna* is. Despite that, not all expressions that satisfy these orthographical requirements are named entities. For example, the expression *EDUCAÇÃO* is not a named entity in the all-uppercase sentence *"CLIQUE AQUI PARA VER A EDUCAÇÃO EM 1993"*, because it is not a proper name. However, a proper name that has been mistakenly written in lowercase will not be considered a named entity according to the guidelines.

2.4 Classification Guidelines

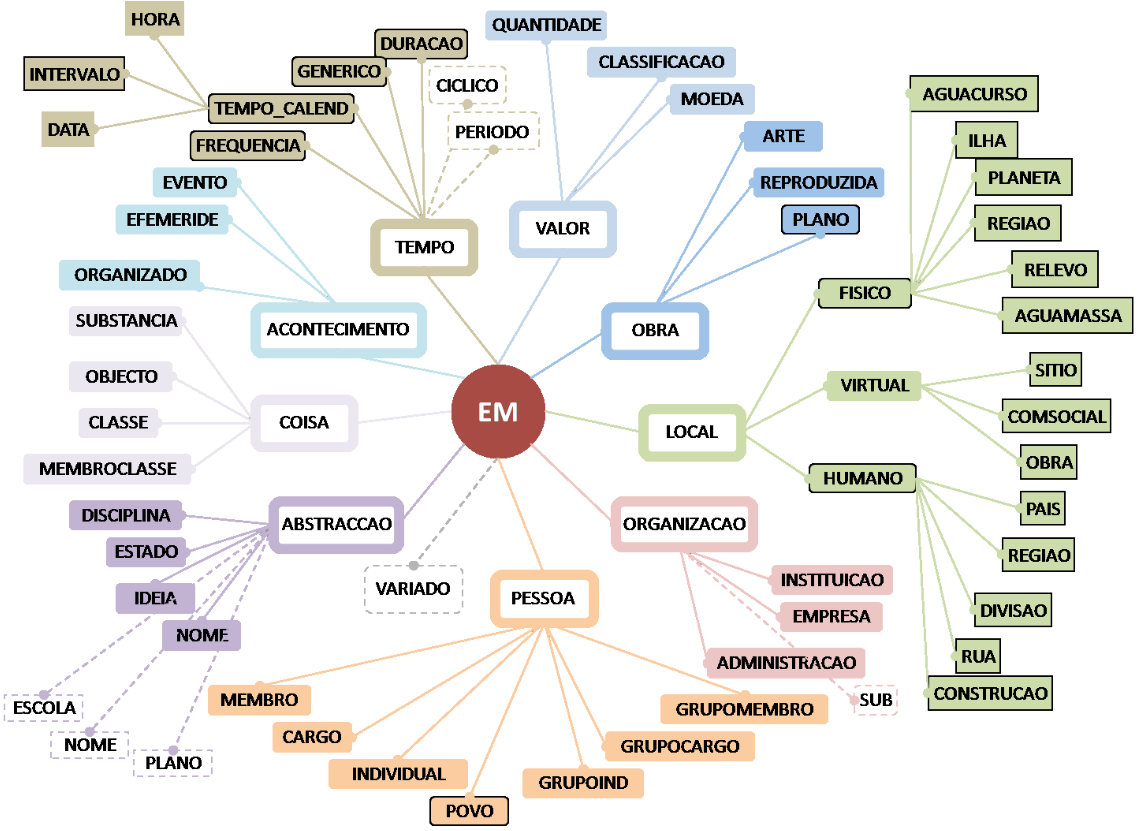
The entire classification hierarchy of the second edition of HAREM can be seen in Figure 2.1. From the innermost boxes to the outermost, these represent, respectively, the categories, the types, and the subtypes, each of which connected to their child classes. Only a few types contain subtypes. The dotted boxes represent classes that were present in the first edition of HAREM, but were removed in the second one. The boxes with black borders represent classes that were introduced only in the second edition. Now we turn to summarize what each category comprehends (SANTOS et al., 2008; CARDOSO; SANTOS, 2007; MOTA et al., 2008a).

2.4.1 PESSOA

2.4.1.1 INDIVIDUAL

Individual persons (e.g. *Miguel de Sá*). If their name is preceded by certain titles, the title is included in the entity as well (e.g. *Dr. Sampaio*, *Presidente da República Jorge Sampaio*). An extensive list of titles that are considered part of an *INDIVIDUAL* entity is given in the guidelines of HAREM. Nicknames (e.g. *Zé*), pet names (e.g. *John (Jack) Reagan)*), initials (e.g. *JFK*), names of mythological and religious entities (e.g. *Deus*), and stand-alone honorifics (e.g. *Vossa Excia*) also belong to this category.

Figure 2.1: Diagram of the class hierarchy of the second edition of HAREM



Source: (CARVALHO et al., 2008)

2.4.1.2 GRUPOIND

Groups of *INDIVIDUAL* entities who, as a group, do not have a fixed name they are known for (e.g. *Vossas Excias*, *Governo Clinton*, casa dos *Mirandas*, o governo de *Cavaco Silva*). A counterexample would be the *Beatles*.

2.4.1.3 CARGO

References to occupations that are being held by certain individual persons at this moment, but that will be held by several others throughout time (e.g. *Papa*, *Ministro dos Negócios Estrangeiros*, *Rainha da Abissínia*). This category also comprehends references to occupations without considering the persons who are behind them (e.g. candidato a *presidente da UE*).

2.4.1.4 GRUPOCARGO

Groups of *CARGO* entities, whether referred to by plural form names or collective names (e.g. *Ministros dos Negócios Estrangeiros da União Europeia*, *Presidência*, *Conselho de Ministros*).

2.4.1.5 MEMBRO

An individual person who is referred to by the organization which they represent (e.g. “Ele foi abordado por um *GNR* à paisana”, “O *Mórmon* estava na sala ao lado”).

2.4.1.6 GRUPOMEMBRO

Groups of *MEMBRO* entities (e.g. “Os *Mórmons* acreditam no profeta John Smith”, “O *BE* reuniu-se ontem”). However a *GRUPOMEMBRO* entity does not possess a personality of its own, such an entity would be better classified as an *ORGANIZATION* (e.g. “O *FC Porto* jogou muito bem e venceu o jogo” contains a *GRUPOMEMBRO*, “O *FC Porto* tem um estádio...” contains an *ORGANIZATION*).

2.4.1.7 POVO

The collection of inhabitants of a location when referred to by the name of that location. (e.g. “Não há música como a do *Brasil*”, “A House Music conquistou a *Inglaterra*, *Holanda*, *Alemanha* e *Ibiza*”, “*Lisboa* ficou horrorizada com essa notícia”).

2.4.2 ORGANIZACAO

2.4.2.1 ADMINISTRACAO

Governmental organizations, such as ministries, municipalities, and chambers (e.g. *Secretaria do Estado da Cultura*, *Brasil*, *Prefeitura de São Paulo*. International and supranational government organizations are also included (e.g. *ONU*, *UE*).

2.4.2.2 EMPRESA

For-profit organizations, such as companies, societies, and clubs (e.g. *Boavista FC*, *Círculo de Leitores*, *Livraria Barata*).

2.4.2.3 INSTITUICAO

Non-profit organizations, such as associations, universities, and political parties (e.g. *Associação de Amizade Portal-Bulgária*, *Universidade Federal do Rio Grande do Sul*, *Liceu Maria Amália*).

2.4.3 LOCAL

2.4.3.1 HUMANO

Human-made locations, such as countries and other geopolitical entities (e.g. *Rio de Janeiro*, *Bairro dos Anjos*, *Ásia Menor*). However a distinction must be made between the government of a location, which is an *ORGANIZATION*, and the location as a spatial concept. The subtypes of the *HUMANO* type are

1. *PAIS*: includes countries, principalities, and unions, such as *União Europeia*.
2. *DIVISAO*: includes cities, villages, and states.
3. *REGIAO*: locations which are product of a cultural or traditional division, with no administrative value, such as *Nordeste*, *Terceiro Mundo*, and *Médio-Oriente*.
4. *CONSTRUCAO*: any kind of building or part of a building, includes bridges, harbors, and pools.
5. *RUA*: any kind of street, road, or square.
6. *OUTRO*: for anything else within this type.

2.4.3.2 FISICO

Locations that were made by nature, such as rivers and mountains. The subtypes of the *FISICO* type are:

1. *AGUACURSO*: bodies of running water, such as rivers and waterfalls.
2. *AGUAMASSA*: bodies of still water, such as lakes and seas.
3. *RELEVO*: land formations, such as mountains and valleys.
4. *PLANETA*: any celestial body.
5. *ILHA*: islands.
6. *REGIAO*: physical geographical regions, such as *Deserto do Sahara*, *Amazonas*,

and the continents.

7. *OUTRO*: for anything else within this type.

2.4.3.3 VIRTUAL

Information spaces, such as the Internet and newspapers. The subtypes of the *VIRTUAL* type are:

- *COMSOCIAL*: any communication medium, such as newspapers, television, and radio.
- *SITIO*: any Internet location, such as websites and FTP sites.
- *OBRA*: any printed work.
- *OUTRO*: for anything else within this type.

2.4.4 OBRA

Any piece of work that is referred to by a proper name.

2.4.4.1 REPRODUZIDA

Mass-produced works, of which there are many copies available, all of them stemming from a single original, such as books and music albums (e.g. *"Turn it on again"*, *"Olhai os Lírios do Campo"*, *Bible*).

2.4.4.2 ARTE

Unique artworks, of which no copies were made, such as monuments and buildings with artistic value (e.g. *Torre Eiffel*, *Cristo-Rei*, *Igreja da Luz*).

2.4.4.3 PLANO

Political, administrative and financial measures, projects, and treaties (e.g *Plano Marshall*, *Orçamento Larou*, and *Rendimento Mínimo Garantido*).

2.4.5 VALOR

Number expressions, with introducing prepositions and quantifiers included (e.g. *cerca de 200 gramas*) as well as number intervals (e.g. *entre 3 e 4%*).

2.4.5.1 CLASSIFICACAO

Ordinal values, values that indicate a position within a sequence, with a subsequent and an antecedent. Establishes things that come after and things that come before, such as classifications, orders, and scores (e.g *2-0*, *3^a*).

2.4.5.2 MOEDA

Currency values, including the currency name or symbol (e.g. *£39*, *50 contos*, *30 milhões de cruzeiros*).

2.4.5.3 QUANTIDADE

Amount values, including the measurement unit if any (e.g. *15 m*, *23%*, *2,500*) . Money amounts are not included, because they already belong to the type *MOEDA*. Some kinds of time amounts are included even though there is a *TEMPO* category for such entities (e.g. “Eu tenho *19 anos*” contains a *QUANTIDADE* entity).

2.4.6 TEMPO

Time expressions, with introducing prepositions and determinants included (e.g. *no ano passado*, *todos os dias*).

2.4.6.1 TEMPO_CALEND

Expressions that address well-defined points in a timeline. The subtypes of the *TEMPO_CALEND* are:

1. *DATA*: represents an absolute date, with full information about the year, the month, and the day to which they refer (e.g. *no dia 19 de Outubro de 2007*), or just information about the year (e.g. *em 1998*). They can also represent a referential date,

i.e., they don't represent a date directly, but they contain a reference through which the absolute date can be known (e.g. *ontem*, *no dia anterior*).

2. *HORA*: represents a time of day (e.g. *às 15:00*).
3. *INTERVALO*: represents a time interval which the limits are *TEMPO* entities (e.g. *entre 2000 e 2003*, *das 12:00 às 14:00 horas*, *de 3 a 6 meses*).

2.4.6.2 DURACAO

Expressions that indicate a time amount (e.g. "Fiquei *dois meses* em Lisboa", "A aplicação da lei será suspensa *por dez anos*").

2.4.6.3 FREQUENCIA

Expressions that indicate the time rate at which an event repeats itself (e.g. "Vou ver os meus pais *todos os dias*", "Vou ver os meus pais *duas vezes por semana*", "Vou ver os meus pais *dia sim dia não*").

2.4.6.4 GENERICO

Expressions that don't fall into any other types, but that nonetheless still indicate time (e.g. "Adoro *o Verão*", "*Fevereiro* é o mês mais curto do ano.").

2.4.7 ACONTECIMENTO

2.4.7.1 EFEMERIDE

A once-in-history event that cannot be repeated (e.g. *Revolução Francesa*, *11 de Setembro*, *2^a Guerra Mundial*).

2.4.7.2 ORGANIZADO

A big organized event, usually containing other smaller *EVENTO* entities (e.g. *Copa*, *Jogos Olímpicos*, *Festival de Jazz do Estoril*).

2.4.7.3 EVENTO

A punctual event, sometimes part of a greater *ORGANIZADO* event (e.g. *Benfica-Sporting*, *Chico Buarque no Coliseu*, *Buzinão na Ponte*).

2.4.8 COISA

2.4.8.1 OBJECTO

An individual thing identified by a proper name. It includes transportation means, individual animals, and planets (e.g. *Titanic*, o cão *Bobi*, *Saturno*).

2.4.8.2 SUBSTANCIA

Noun-individualized material stuff, usually chemical substances (e.g. *HDL*, *vitamina B12*, *CO2*).

2.4.8.3 MEMBROCLASSE

Individual things that are referred to by the name of the class to which they belong (e.g. “O meu *Fiat Punto* foi à revisão”, O *MS Word 2003* da Cristina rebentou hoje”).

2.4.8.4 CLASSE

A collection of things that is referred to by a single name, such as brands, models, pedigrees, and computer software (e.g. “A FCN exige relatórios em folhas *A4*, “Os móveis *Luís XV* são muito raros”).

2.4.9 ABSTRACCAO

2.4.9.1 DISCIPLINA

Scientific disciplines, theories, technologies and practices (e.g. *Inteligência Artificial*, *Teoria da Relatividade*, *Tai-Chi*).

2.4.9.2 ESTADO

Physical states, conditions, functions, and diseases (e.g. *doença de Alzheimer*, *AIDS*, *Sistema Nervoso Central*).

2.4.9.3 IDEIA

Abstract things, like ideas and ideals (e.g. "O senhor acredita na *Ressurreição*?", "Qualquer dia já ninguém acredita na *República* e na *Democracia*.", "Neste blogue praticam-se a *Liberdade* e o *Direito de Expressão* próprios das sociedades avançadas").

2.4.9.4 NOME

A name in itself, without relation to the thing that it names (e.g. "Achei um cão. Vou dar-lhe o nome de *Bobi*", "O magnata criou uma empresa chamada *Cauca7*").

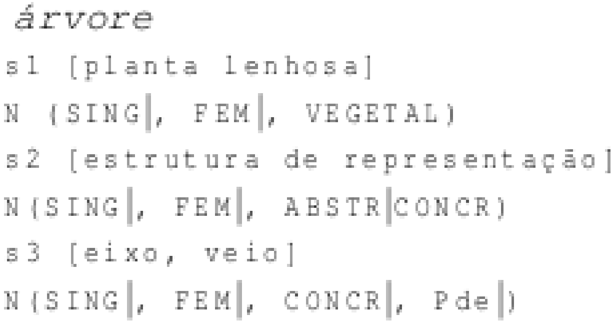
3 RELATED WORK

In this chapter we present a series of works directly related to the proposal of this paper. We start by giving an overview of the strategy used by each of the three best participants of the second edition of HAREM, namely, Priberam, REMBRANDT, and XIP. Then, we contrast those systems with the work of Chiu and Nichols (2015), who proposed a Bidirectional LSTM-CNN system for solving the named entity recognition task of CoNLL-2003.

3.1 Priberam

Priberam (AMARAL et al., 2008) was the participant in HAREM that achieved the highest F-score on the classical track, 57.11%. The system was supported by a lexicon that contained, for each entry, a list of its possible meanings, and, for each meaning, its part-of-speech tag and its position inside an ontology. An example of such an entry is seen in Figure 3.1.

Figure 3.1: Priberam's lexicon entry



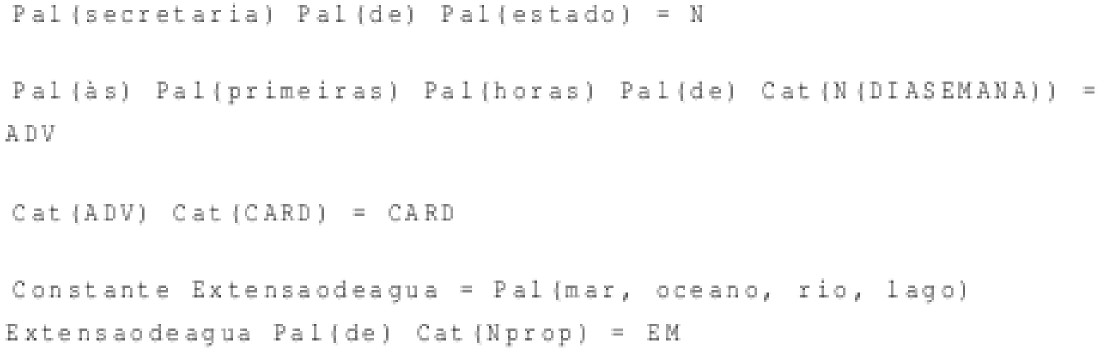
```
árvore
s1 [planta lenhosa]
N (SING|, FEM|, VEGETAL)
s2 [estrutura de representação]
N(SING|, FEM|, ABSTR|CONCR)
s3 [eixo, veio]
N(SING|, FEM|, CONCR|, Pde|)
```

Source: (AMARAL et al., 2008)

Here the word *árvore* is given three possible meanings and, since all of them are classified as nouns, further information is given about their grammatical number, grammatical gender, and the ontological supercategories they belong to. However, the lexicon provides information only for words in isolation, information about entire expressions is inferred through context rules. Some examples of these rules are seen in Figure 3.2.

The rules allow a part-of-speech tag to be attributed to an entire sequence of exact words or word categories. The first rule in the example shows a sequence of *Pal* units, which stand for the exact word given as argument, that wherever are found in the

Figure 3.2: Priberam's context rules



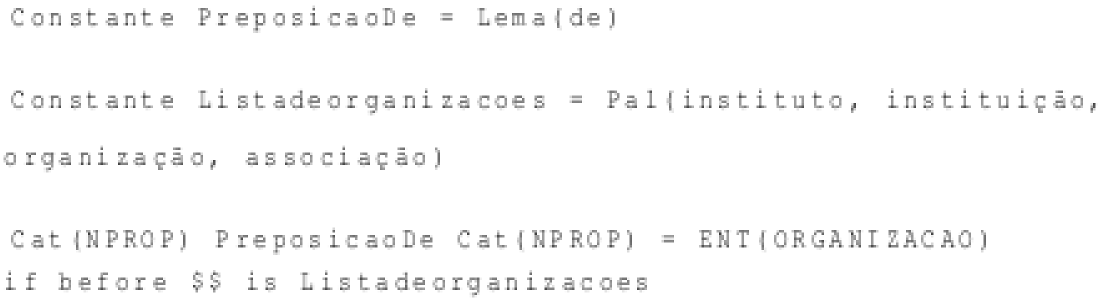
```
Pal(secretaria) Pal(de) Pal(estado) = N

Pal(às) Pal(primeiras) Pal(horas) Pal(de) Cat(N(DIASEMANA)) =
ADV

Cat(ADV) Cat(CARD) = CARD

Constante Extensaodeagua = Pal(mar, oceano, rio, lago)
Extensaodeagua Pal(de) Cat(Nprop) = EM
```

Source: (AMARAL et al., 2008)

text are to be treated as a single noun. The second rule presents the *Cat* unit, which stands for any word that belong to the category given as argument, in this particular case *Cat(N(DIASEMANA))* represents any noun that means a day-of-week. The third rule showcases that an expression can be made entirely of high-level category units, without any reference to an exact word. Finally, the fourth rule describes the *Constante* construct, which basically defines an alias to replace a given expression and defines a category for it, in this case *EM*, which stands for named entity. All these rules are applied recursively to each other until there are only *Pal* units, that are finally matched against expressions in the text.

The *Constante* construct is very helpful in describing in a concise way the most formulaic named entities, such as the ones that are preceded by certain key words that give away the their category. Figure 3.3 shows an example.

Figure 3.3: Priberam's constant constructions



```
Constante PreposicaoDe = Lema(de)

Constante Listadeorganizacoes = Pal(instituto, instituição,
organização, associação)

Cat(NPROP) PreposicaoDe Cat(NPROP) = ENT(ORGANIZACAO)
if before $$ is Listadeorganizacoes
```

Source: (AMARAL et al., 2008)

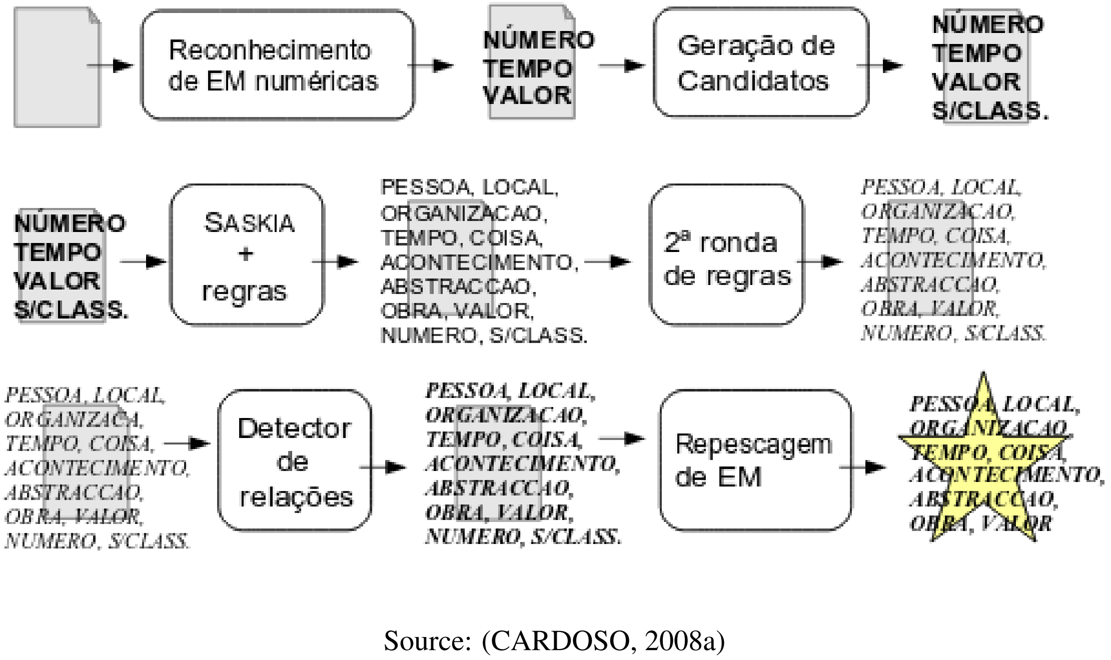
In this example, the constant *Listadeorganizacoes* unites under a single alias several words with similar meanings that often are related to *ORGANIZACAO* entities.

3.2 REMBRANDT

REMBRANDT (CARDOSO, 2008a) was the participant of HAREM with the second best F-score achieved, 56.74% on the classical track. It was made for Portuguese and is capable of recognizing named entities and detecting relationships between them. The system works by applying on named entities grammar rules that take advantage of internal and external hints for extracting their meaning. REMBRANDT has its origins on the task of annotation of geographical named entities. It aimed to establish for documents their geographical signature, i.e., the geographical scope the documents worked within. REMBRANDT is based on PALAVRAS, the same morphosyntactical analyzer that was used by the best participant of the first edition of HAREM, and uses the Wikipedia as a source of extra information about the named entities.

The recognition of named entities happens in three phases, as described in Figure 3.4. Each row of the figure represents a phase.

Figure 3.4: Priberam's constant constructions


Reconhecimento de EM numéricas
NÚMERO TEMPO VALOR
Geração de Candidatos
NÚMERO TEMPO VALOR S/CLASS.
NÚMERO TEMPO VALOR S/CLASS.
SASKIA + regras
PESSOA, LOCAL, ORGANIZACAO, TEMPO, COISA, ACONTECIMENTO, ABSTRACCAO, OBRA, VALOR, NUMERO, S/CLASS.
2ª ronda de regras
PESSOA, LOCAL, ORGANIZACAO, TEMPO, COISA, ACONTECIMENTO, ABSTRACCAO, OBRA, VALOR, NUMERO, S/CLASS.
PESSOA, LOCAL, ORGANIZACA, TEMPO, COISA, ACONTECIMENTO, ABSTRACCAO, OBRA, VALOR, NUMERO, S/CLASS.
Detector de relações
PESSOA, LOCAL, ORGANIZACAO, TEMPO, COISA, ACONTECIMENTO, ABSTRACCAO, OBRA, VALOR, NUMERO, S/CLASS.
Repescagem de EM
PESSOA, LOCAL, ORGANIZACAO, TEMPO, COISA, ACONTECIMENTO, ABSTRACCAO, OBRA, VALOR

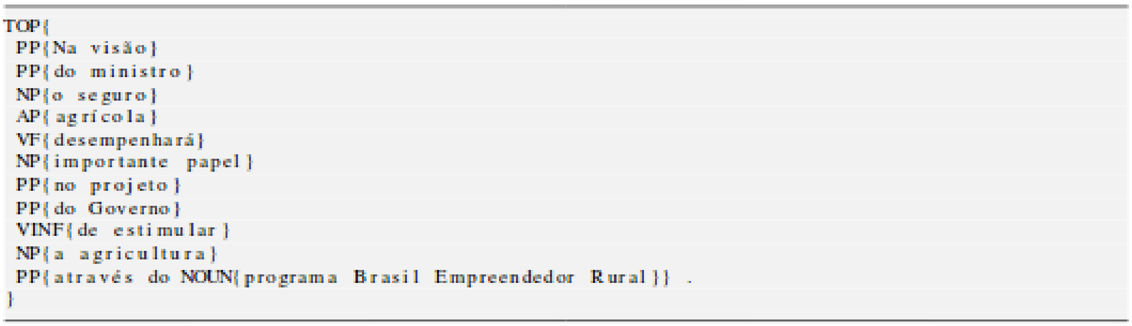
Source: (CARDOSO, 2008a)

In phase 1, the number expressions are recognized and candidates for named entities are generated. First, the input text is divided into sentences and tokens and then all number expressions are identified. Using the knowledge acquired from the already identified number expressions, time expressions and values are identified next. Finally, a set of candidate named entities is generated by observing the presence of any uppercase letters or digits inside the expressions, with the addition of any so-called *daeose* expressions. These expressions are the ones formed by the words *de*, *da*, *do*, *das*, *dos*, and *e*, as long as they are not situated at the borders of the named entity candidate. In phase 2, the named entities are classified with the help of SASKIA, an interface to Wikipedia. SASKIA collects the different meanings a named entity can have by using the disambiguation pages from Wikipedia. The work of SASKIA is also supervised by a series of grammar rules that look for internal and external hints in the named entity. Those named entities that contain *daeose* expressions are checked for the possibility of being attributed an *ALT* tag or to be broken down in smaller entities and then be classified again. In phase 3, the unclassified named entities are captured. This is the moment that rules for detecting relationships between named entities are applied—a task not evaluated on the classical track—, and the results are used to recapture unclassified named entities that happen to be related to already classified ones. Also, a list of common people's names is used for recognizing remaining person entities. If there are still unclassified named entities, those are eliminated, as well as numbers written in full without any uppercase letter.

3.3 XIP

XIP (HAGèGE; BAPTISTA; MAMEDE, 2008) had the third best F-score on the classical track of HAREM, 54.45%. The system was developed as a partnership between Xerox and the L^{2}F from INESC-ID Lisboa. It is a tool for lexical, syntactical, and semantic processing primarily made for extracting syntactical dependencies. Figures 3.5 and 3.6 show the result of applying, respectively, XIP's chunker and dependency tree parser to the sentence *"Na visão do ministro, o seguro agrícola desempenhará importante papel no projeto do Governo de estimular a agricultura, através do programa Brasil Empreendedor Rural"*.

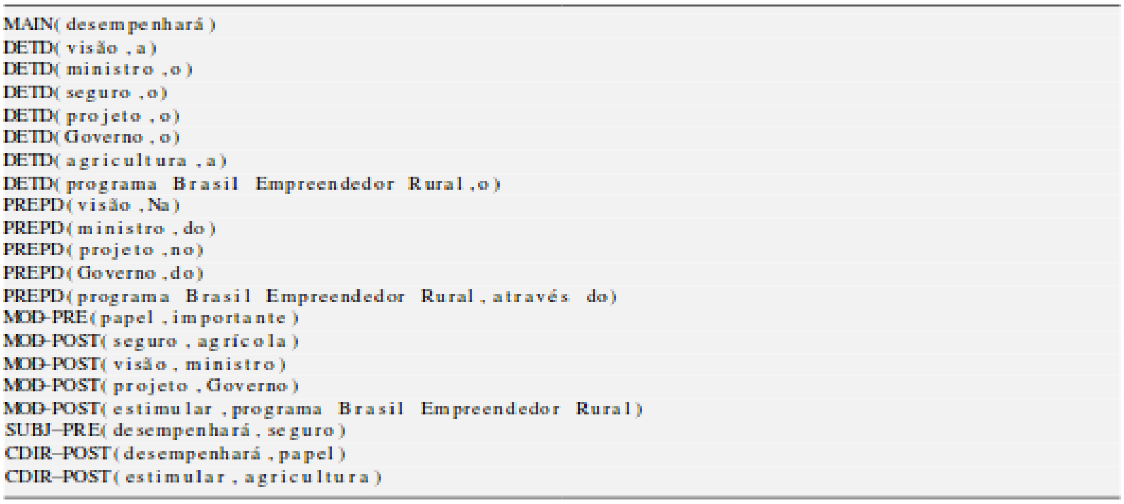
Figure 3.5: XIP's chunker



```
TOP{
 PP{Na visão}
 PP{do ministro}
 NP{o seguro}
 AP{agrícola}
 VF{desempenhará}
 NP{importante papel}
 PP{no projeto}
 PP{do Governo}
 VINF{de estimular}
 NP{a agricultura}
 PP{através do NOUN{programa Brasil Empreendedor Rural}} .
}
```

Source: (HAGèGE; BAPTISTA; MAMEDE, 2008)

Figure 3.6: XIP's dependency tree parser



```
MAIN( desempenhará )
DETD( visão , a )
DETD( ministro ,o )
DETD( seguro ,o)
DETD( projeto , o)
DETD( Governo , o)
DETD( agricultura , a )
DETD( programa Brasil Empreendedor Rural ,o )
PREPD( visão ,Na )
PREPD( ministro , do )
PREPD( projeto ,no)
PREPD( Governo ,do)
PREPD( programa Brasil Empreendedor Rural , através do)
MOD-PRE( papel , importante )
MOD-POST( seguro , agrícola )
MOD-POST( visão , ministro )
MOD-POST( projeto , Governo )
MOD-POST( estimular , programa Brasil Empreendedor Rural)
SUBJ–PRE( desempenhará , seguro )
CDIR–POST( desempenhará , papel )
CDIR–POST( estimular , agricultura )
```

Source: (HAGèGE; BAPTISTA; MAMEDE, 2008)

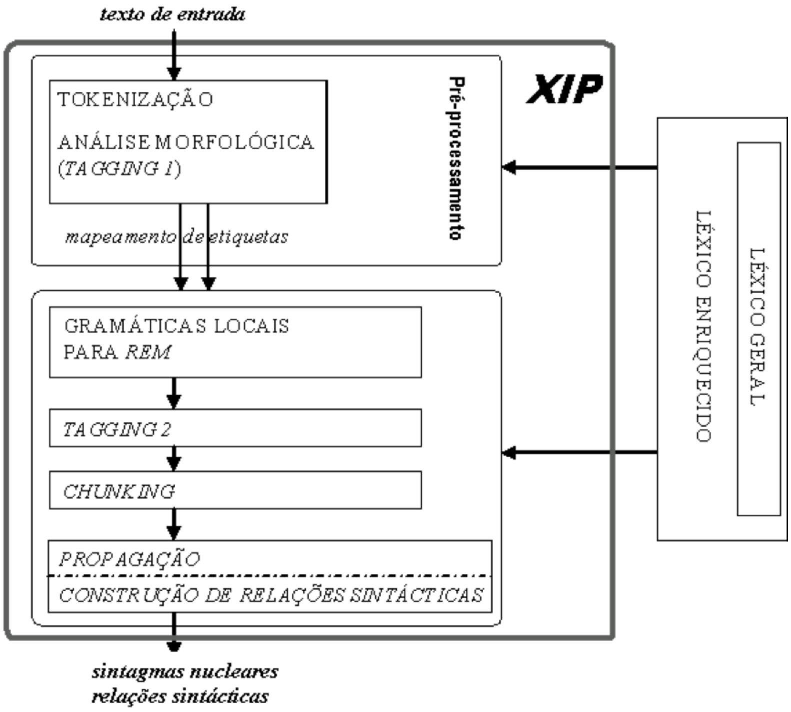
The module for named entity recognition was a later addition to this larger system for syntactical analysis. This integration was motivated by the fact that both portions of the system would help each other in their functions. For instance, sometimes named entities have a complex syntactical structure on the surface, but in reality they correspond to just a name. Consider the expression *"E tudo o vento levou"*, which has the structure of a complete sentence, but, when it refers to the title of the film, it has the value of a name.

1. Fomos rever *E tudo o vento levou* ontem.

On the other hand, by taking advantage of the syntactical structure of the text sometimes it is possible to disambiguate the category of a named entity. For instance, in the sentence *"Portugal respondeu..."*, the named entity *Portugal* does not mean a place and more likey means a person. That is because it is the subject of a verb, which is not the position places occupy.

The general architecture of XIP is presented in Figure 3.7. An entry in lexicon is a set of key-value pairs. The keys are *lemma*, *surface*, *maj*, *toutmaj*, and other custom keys defined in the grammar. The values of *lemma* and *surface* are strings, and the values of *maj* and *toutmaj* are booleans indicating, respectively, whether the surface form begins with capital letter and whether the surface form is completely written in uppercase. There are two types of lexicons in XIP, the preexisting one and the custom defined one. The preexisting lexicon comes from the results of the POS tagging tool—this step is also called syntactical preprocessing. The custom defined lexicon can have its entries defined

Figure 3.7: XIP's architecture



Source: (HAGèGE; BAPTISTA; MAMEDE, 2008)

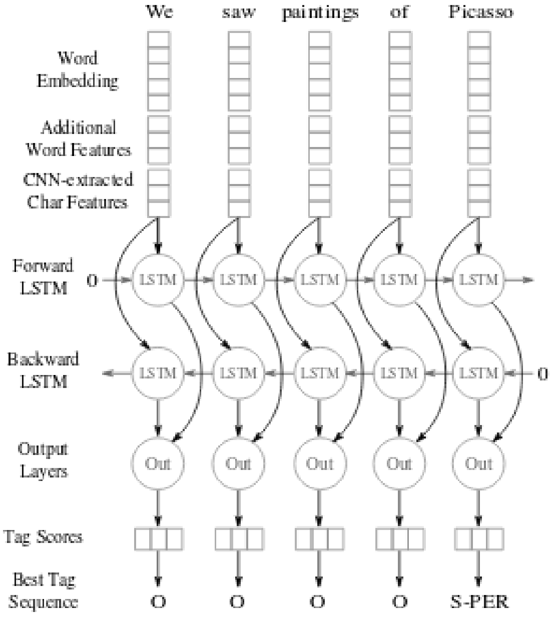
directly or be made out of changes in the already existing lexical entries. Moreover, rules for category disambiguation are applied in three steps, Tagging 1, Tagging 2, and a final selection that is done by a Hidden Markov model. At the end of this process, entities such as *Natal* can be distinguished by whether they refer to a place or an event. The local grammar module has also rules for considering the contexts to the right and to the left of an expression, and so be able to join multiword named entities. In the last phases of processing, XIP then uses the information of chunking—since many named entities are nouns—and syntactical dependencies—to deal with metonyms—for refining the results. A distinctive feature of XIP is the use of propagation. This is a mechanism for conserving information of environments where a named entity is richly characterized to transpose this knowledge to environments where entities are poorly characterized. The drawback is that if a named entity was misclassified at first, then this mistake will be propagated to other environments. In HAREM, the use of this feature was restricted only to the category *PESSOA*.

3.4 Bidirectional LSTM-CNN

Chiu and Nichols (2015)'s Bidirectional LSTM-CNN system was not a participant of HAREM, instead of this, it was evaluated according to the guidelines of CoNLL-2003. The system is an improvement over the work of Collobert and Weston (2008) , who proposed a feed-forward neural network to solve the named entity recognition task. However his approach fails in two ways: it restricts the use of context to a fixed-sized window around each word, and it depends solely on words embeddings, leaving out rare words. Chiau et al solves the first problem by using a bidirectional LSTM, and the second one by using character-level features.

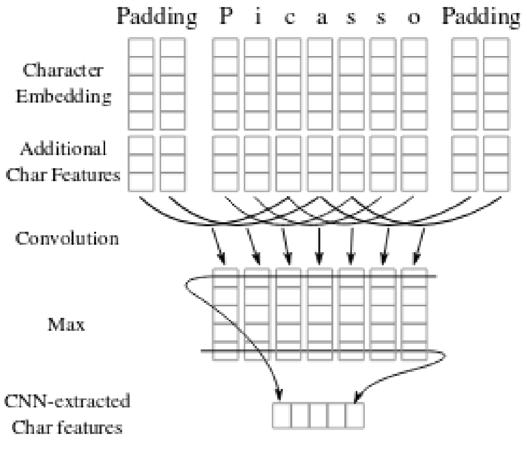
The architecture of the system is shown in Figure 3.8. It starts by converting each token of the text to its correspondent word embedding. Word embeddings are representations of discrete features, such as words and characters, in the form of continuous vectors—a more suitable format to be inputted to a neural network. Besides that, the resulting vector is concatenated to a vector of additional word features. These features indicate information about graphical aspects of the word and can assume the following options: *allCaps*, *upperInitial*, *lowercase*, *mixedCaps*, *noinfo*. Finally, it is appended to the neural network input a last vector of character features obtained from the CNN module. The module is shown in Figure 3.9 in more detail. This module is responsible for figuring out morphological characteristics of the words—such as the presence of certain prefixes and suffixes—that can help in recognizing it as a named entity. It works by first padding each word on both sides with a special *PADDING* character so as to make all words the same length, and then converting them to character embeddings. These embedding are randomly initiated with values drawn from a uniform distribution. Furthermore, a vector of additional character features is also added indicating whether each character is uppercase, lowercarse, punctuation or other. The input vectors are then passed through a convolution and a max layer to generate the character features vectors. Back to the calling module, the resulting input vectors containing the word embeddings, the additional word features and the CNN-extracted character features is passed to a forward and backward LSTM layer. These layers will take information from both past and coming words to build a context for the current word being analyzed. The system will finally output a continuous value that can be discretized into a score tag for each category. The tag that receives the best score is selected as the solution for the classification of that word. The tag scheme used was BIOES with the CoNLL-2003 named entity categories. BIOES stands for *Begin*, *Inside*, *Outside*, *End*, and *Single* and those are the possible tags supported by the scheme. The tag *O* is used for words that are not within a named entity expression; the tag *S* are for words that are within named entity expressions and are the only ones to be so; the tags *Begin*, End, and *Inside* are for words that are, respectively, at the beginning, at the end, and at a middle position of a named entity expression. The category of the named entity is expressed by appending its name to a non-*O* tag, e.g. *S-PER* for a word that singly composes a named entity and is from the category PER.

Figure 3.8: Architecture of Bidirectional LSTM-CNN neural network



Source: (CHIU; NICHOLS, 2015)

Figure 3.9: CNN module of Bidirectional LSTM-CNN neural network



Source: (CHIU; NICHOLS, 2015)

The results of the system across different models is shown in Table 3.1. The first model is a feed-forward neural network (FFNN) improved with word embeddings (emb), capitalization features (caps), and a lexicon (lex). The second model is a bidirectional LSTM (BLSTM) alone. The third to fifth models are hybrid bidirectional LSTM and CNNs improved with, respectively, nothing, word embeddings, and word embeddings and a lexicon. The results are given in terms of precision (Prec.), recall, and F-score (F1, standard deviation given in parenthesis). The best F-score result was achieved with the BLSTM-CNN + emb + lex model, 91.62%, however this is very close to the result achieved without the use of a lexicon, 90.91%. When considering the cost of handling the lexicon, the BLSTM-CNN + emb model is more advantageous.

Table 3.1: Results of different models of the Bidirectional LSTM-CNN architecture

Model	**CoNLL-2003**		
	Prec.	**Recall**	**F1**
FFNN + emb + caps + lex	89.54	89.80	89.67 (±0.24)
BLSTM	80.14	72.81	76.29 (±0.29)
BLSTM-CNN	83.48	83.28	83.38 (±0.20)
BLSTM-CNN + emb	90.75	91.08	90.91 (±0.20)
BLSTM-CNN + emb + lex	91.39	**91.85**	**91.62** (±0.33)

Source: (CHIU; NICHOLS, 2015)

4 METHODS

4.1 Bidirectional LSTM

A bidirectional LSTM (SCHUSTER; PALIWAL, 1997) is a kind of recurrent neural network. The distinguishing characteristic of such networks is that their outputs are fed again to its inputs—alongside other new ones—at the end of each cycle. That allows conclusions from the past to influence the present, creating a sense of time ordering in the system, something that does not happen in simple feed-forward networks. LSTMs (short for *Long Short Term Memory*) are special because they are able to keep information from cycles that happened much further back in the past by means of gated cells. These are cells that are guarded by so-called gates, mechanisms that decide whether the cell will receive a new value, or the cell will pass on its current value, or the cell will have its value forgotten. This process allows certain values to be retained within gated cells for many cycles until they are free to interact with the rest of the system, and, through the forgetting feature, it is possible to guarantee that the system as a whole will be balanced with new and old information. A bidirectional LSTM is basically a pair of a forward LSTM, that works from the past to the future, and a backward LSTM, that works in the opposite direction. With the interaction of both of them we have a system which the current output is influenced by both the past and future inputs. Therefore, a bidirectional LSMT requires as input a vector of timesteps representing each position in a timeline. In the case of Chiu and Nichols (2015)'s system, the notion of timestep is translated as each word input vector of a sentence. So given a word, the future words are the words to the right of it, and the past words are the words to the left. Together they form the context which is considered when deciding the corresponding output tag for that central word.

4.2 CNN

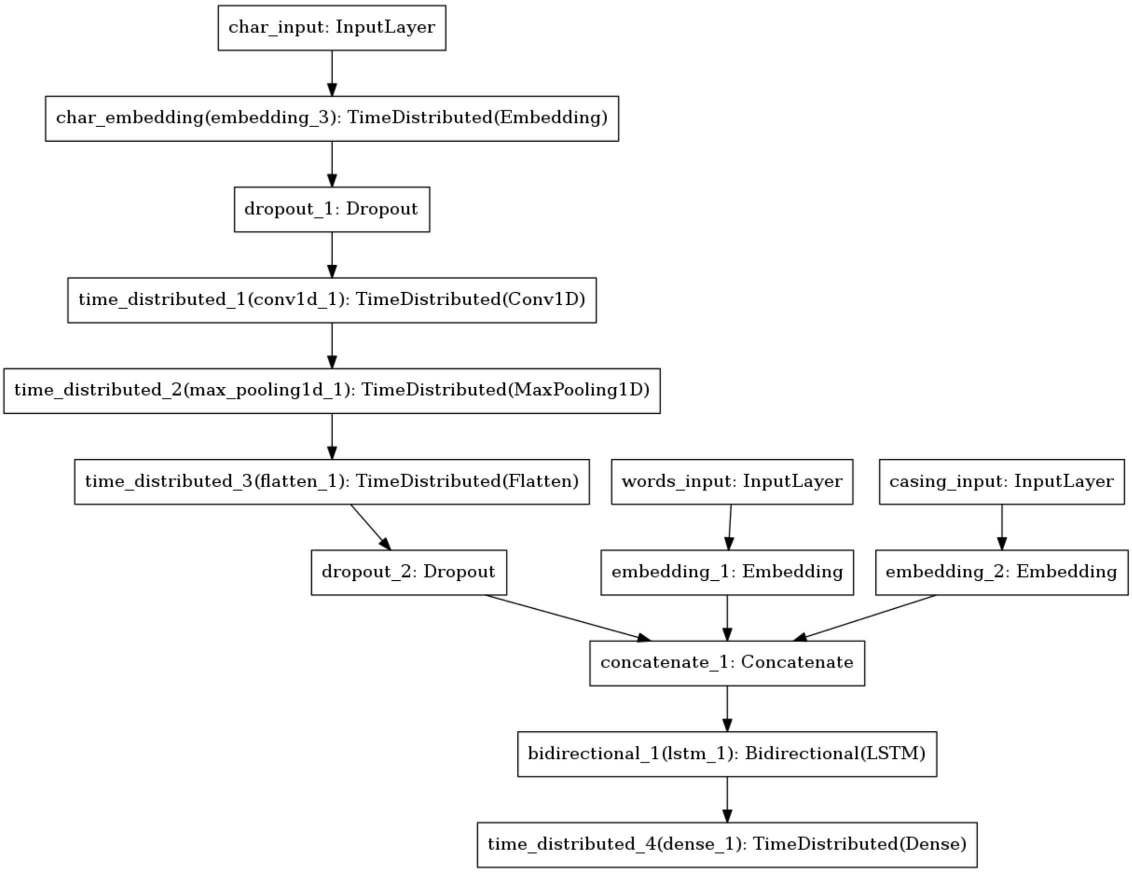
CNNs (short for *Convolutional Neural Network* are neural networks that resemble the way the human vision works when making out features from contiguous regions of visual input (KARPATHY, 2017). This window that selects each region to be currently considered is named a kernel and it moves across the input area one stride at a time until covering everything. Every time the kernel is over a region, it determines a cell of a feature map, a representation of more abstract notions the network is reaching from the concrete input. The borders of the input area are padded to avoid a decreasing in the size of the feature map at every iteration. To make the feature map advance in the abstractness of its representation, a pooling layer is used to shrink it. In Chiu and Nichols (2015)'s system, the input is the matrix formed by the character embeddings with additional features added, and the word characters.

4.3 Raj (2018)'s implementation

In this work, we consider the Bidirectional LSTM-CNN system as implemented by Raj (2018) . This is an implementation based on Keras (CHOLLET et al., 2015) of Chiu and Nichols (2015)'s system that differs from the original paper in the following ways: (1) it does not use lexicons; (2) it uses bucketing to speed up the training; and (3) uses *nadam* optimizer instead of *SGD*. It reports achieving an F-score of 90.9% with approximately 70 epochs against the F-score of 91.4% of the original paper for the same architecture, i.e., with word embeddings and capitalization features.

A scheme of the Keras model used for this implementation is shown in Figure 4.1. Before passing through the neural network, the input data goes through a preprocessing stage. It starts by receiving as training input a corpus file formatted in the CoNLL-2003 standard: each line represents a word record and the end of a full sentence is represented by a blank line. Each word record contains in the first column the word, in the second column the POS tag, in the third column the syntactic chunk tag, and in the last column the named entity tag in the BIO format. Only the first and last column are actually used by the system, the rest being filtered out. It proceeds to creating a vector out of the characters that make up each word and adding those to the input vector alongside the original words. Then it creates a mapping to each unique word (with case ignored), output label, case type, and character to an index, so as to make it easier for the network to handle the data. A case type is a word feature that can assume the following values: *numeric*, *allLower*, *allUpper*, *initialUpper*, *other*, *mainly_numeric*, *contains_digit*, and *PADDING_TOKEN*. The mapping of characters contain all characters from the charset plus a padding character and an unknown character—for characters outside of the charset. Next, the system creates an embedding for the case types consisting of an identity matrix and an embedding for the words based on the 100-dimensional GloVe (PENNINGTON; SOCHER; MANNING, 2014) word embeddings plus a zero vector for a PADDING_TOKEN and a random vector from the interval $[-0.25, 0.25]$ for an UNKNOWN_TOKEN. With the mappings created, the input matrix is generated as a a concatenation of word indices, case indices, character indices, and label indices of each input word. The character indices are padded in such a way as to make all of them have 52 characters of length. Finally, the input matrix is broken into batches of same-sized words to speed up the training stage, since in this way it is possible to train them in parallel.

Figure 4.1: Raj (2018)'s Bidirectional LSTM-CNN model


char_input: InputLayer
char_embedding(embedding_3): TimeDistributed(Embedding)
dropout_1: Dropout
time_distributed_1(conv1d_1): TimeDistributed(Conv1D)
time_distributed_2(max_pooling1d_1): TimeDistributed(MaxPooling1D)
time_distributed_3(flatten_1): TimeDistributed(Flatten)
words_input: InputLayer
casing_input: InputLayer
dropout_2: Dropout
embedding_1: Embedding
embedding_2: Embedding
concatenate_1: Concatenate
bidirectional_1(lstm_1): Bidirectional(LSTM)
time_distributed_4(dense_1): TimeDistributed(Dense)

Source: (RAJ, 2018)

In the next stage, the input matrix is given to the neural network per se. The models has three inputs, *char_input*, *words_input*, and *casing_input* that are concatenated midway and then output a label. An explanation of each layer is given:

1. **char_input**: the character vector input.
2. **char_embedding(embedding_3)**: character embeddings initialized from a random uniform distribution in the interval $[-0.5, 0.5]$. The dimension of the output is 30.
3. **dropout_1**: dropout layer with a rate of 50%.
4. **time_distributed_1(conv1d_1)**: the layer of convolution. The length of the convolution window is 3, the dimensionality of the output filter is 30, the input is padded in such a way that the output has the same length as the originial input, the length

of the stride is 1, and the activation function is tanh.

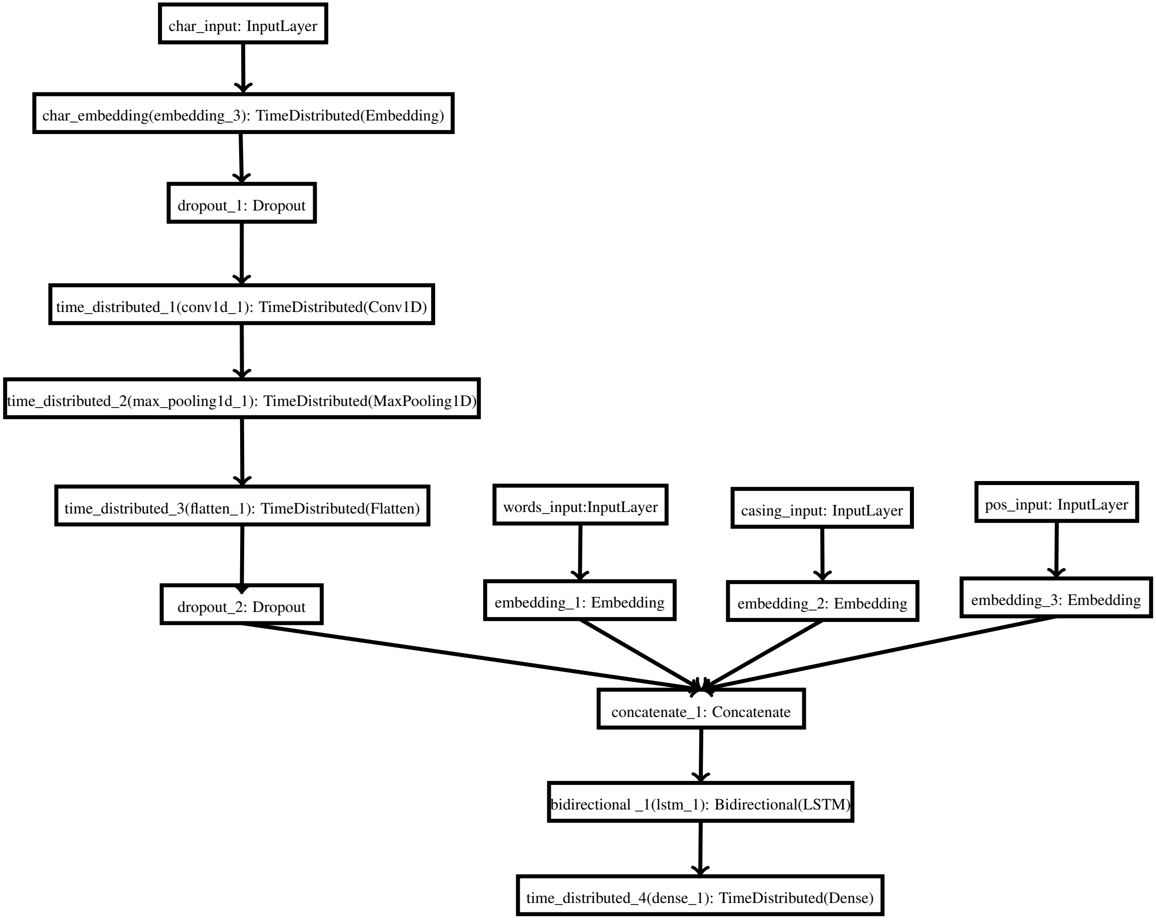
5. **time_distributed_2(max_pooling1d_1)**: the max pooling layer with size of window 52.
6. **time_distributed_3(flatten_1)**: flattening of the max pooling layer.
7. **dropout_2**: dropout layer with a rate of 50%.
8. **words_input**: the word vector input.
9. **embedding_1**: word embeddings as defined in the preprocessing stage.
10. **casing_input**: the case vector input.
11. **embedding_2**: case embeddings as defined in the preprocessing stage.
12. **concatenate_1**: the layers words, casing, and char are concatenated as a single input to the next layer.
13. **bidirectional_1(lstm_1)**: the bidirectional LSTM layer, with 200 units of input, dropout of 50%, and recurrent dropout of 50%.
14. **time_distributed_4(dense_1)**: the last activation layer with a softmax function. The model is compiled with loss function of *sparse_categorical_cross_entropy* and optimizer *nadam*.

4.4 HAREM adaptation

Our goal was to adapt Raj (2018)'s implementation to the HAREM environment and in accordance to the guidelines of the contest[1] . For this, we built a module for translating HAREM's input file format to one closer to CoNLL's, the one handled natively by the original implementation. HAREM's input file consists of a series of documents identified by DOC elements structured in an XML syntax. We ignored OMITIDO tags, that refer to text that can be safely ignored because they are either not written in Portuguese or they are written in ungrammatical language. In cases of vagueness, we chose to always use the first alternative given since the system was not originally designed to deal with multiple outputs. Our module then transforms the HAREM input to something akin to CoNLL input—an intermediate format—, a document with one word per line, sentences being delimited by a blank line, and each line containing the surface form of the word and the named entity BIO label. The BIO label is slightly different because HAREM has different categories. We still used the BIO scheme from CoNLL, but now with the names

[1] Source code for the implementation is available at <https://gitlab.com/concys/ner-harem>

Figure 4.2: Model of HAREM adaptation enriched with POS-tagging data


char_input: InputLayer
char_embedding(embedding_3): TimeDistributed(Embedding)
dropout_1: Dropout
time_distributed_1(conv1d_1): TimeDistributed(Conv1D)
time_distributed_2(max_pooling1d_1): TimeDistributed(MaxPooling1D)
time_distributed_3(flatten_1): TimeDistributed(Flatten)
words_input:InputLayer
casing_input: InputLayer
pos_input: InputLayer
dropout_2: Dropout
embedding_1: Embedding
embedding_2: Embedding
embedding_3: Embedding
concatenate_1: Concatenate
bidirectional _1(lstm_1): Bidirectional(LSTM)
time_distributed_4(dense_1): TimeDistributed(Dense)

Source: The Authors

of the HAREM categories added to it. We opted for not go further than the category level in the classification of named entities, therefore we do classify them into types and subtypes. This decision was motivated by our belief that such a complex classification hierarchy would confuse the system. Since HAREM input documents do not natively divide sentences, we used CoreNLP (MANNING et al., 2014) to achieve this task. The same tool was also used for word tokenization. For word embeddings, we used 100-dimensional GloVe embeddings for Portuguese from the NILC repository (HARTMANN et al., 2017). For the character embeddings, we changed the charset to windows-1252, the one used by HAREM. At the end of the whole process, we convert the output file from the intermediate format back to the HAREM format, giving the named entities incremental ID attributes.

The platform used to run the experiments was an Intel(R) Core(TM)2 Quad CPU Q8200 @ 2.33GHz with 4096 kB of RAM. The operating system was a Ubuntu 18.04.1 LTS on a Linux 4.15.0-36-generic kernel. The software was coded in Python 3.6.6 and the neural network model was made with Keras 2.2.2.

4.5 HAREM adaptation enriched with POS-tagging data

We followed the trend of other HAREM participants, such as XIP, of considering the influence of the morphological class of a word in its status as named entities. Named entities are nouns, therefore we hypothesized that knowing whether a certain word is a noun could improve the performance of the HAREM-adapted system. For this, we used CoreNLP POS-tagging module trained for Portuguese (TOUTANOVA et al., 2003; CORTES, 2018) when converting the HAREM input format to our intermediate format, or when doing the opposite with the output side. In this way, the intermediate format gained a new column informing for each word its POS tag, alongside the old columns of surface form and named entity BIO label. Figure 4.2 shows the neural network model of this implementation, where we added a new input branch very similar to the one started by *casing_input*. The POS-tagging embeddings are made by building an identity matrix with each row-column representing a different morphological class available by the CoreNLP module. At the end of this new POS-tagging input, the flow of data is concatenated with the word input, the character input and the casing input and proceeds to the bidirectional LSTM layer.

5 RESULTS

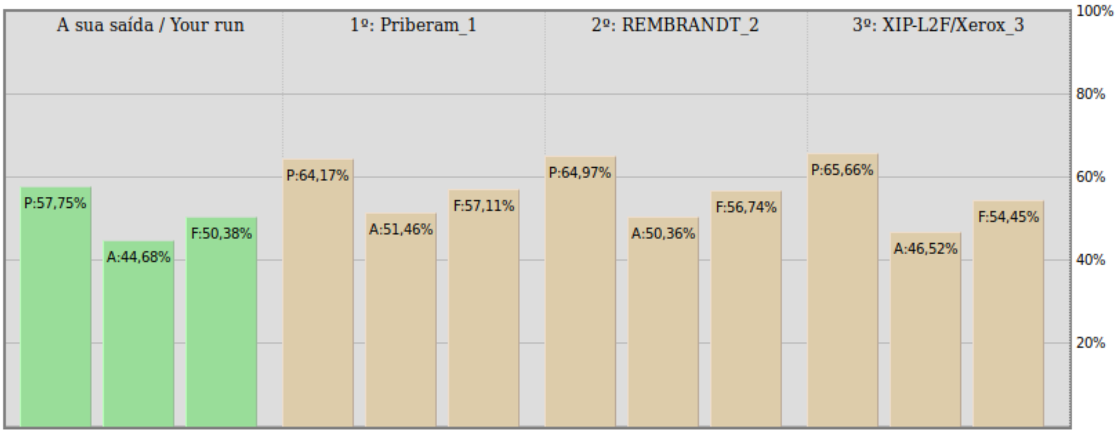
We evaluated the performance of our system on SAHARA (CARDOSO, 2008b). This platform automatically evaluates a given output in HAREM format according to the guidelilnes of the second edition of the contest. The testing corpus was the Golden Collection of the Second HAREM (MOTA et al., 2008b), a compilation of several documents in Portuguese language from a broad range of subjects, and the training corpus was a Second HAREM compatible version of the First HAREM's golden collection. This is the standard way of testing a system as defined by HAREM. SAHARA then outputs a graph showing the precision (marked as *P*), recall (marked as *A*), and F-score (marked as *F*) of the user-given system against those of Priberam, REMBRANDT, and XIP-L2F/Xerox. SAHARA was run with the default options, which makes it evaluate the system on the classical track, the most general track of named entity recognition in the contest.

The results from running the Bidirectional LSTM-CNN system in HAREM are shown in Figure 5.1. It achieved an F-score of 50.38%, staying behind all the other participants, but still very close to XIP's performance, with a difference of 4.07%. It also achieved a precision of 57.75% and a recall of 44.68%, which, in a similar manner, were not enough to surpass any other of the three best participants, but were close to. The difference between this system and the participant with the closest precision—Priberam—was 6.42%, and the difference between this system and the participant with the closest recall—XIP—was 1.84%.

When the system is run with the addition of POS-tagging data, its results become as shown in Figure 5.2. The F-score went from 50.38% to 50.67%, an increase of 0.29%, which was not significant enough to change the rank of the system in relation to its competitors. The precision suffered an increase of 1.17% and the recall suffered an increase of 0.29%, making them, respectively, 54.64% and 45.85%.

Therefore, the results point that the HAREM-adapted Chiu and Nichols (2015)'s system with no POS-tagging addition is not a powerful enough candidate to overcome the best systems based on hand-crafted rules, however it is a very close match to them. Considering that, in contrast to the other participants, this system is based on a deep learning model and it was not originally designed for Portuguese, it shows that it is possible to achieve good results with this approach. The results also reveal that adding POS-tagging data does not improve significantly the performance of the system. A hypothesis for the cause of this behavior may be the fact that the neural network ends up figuring out the

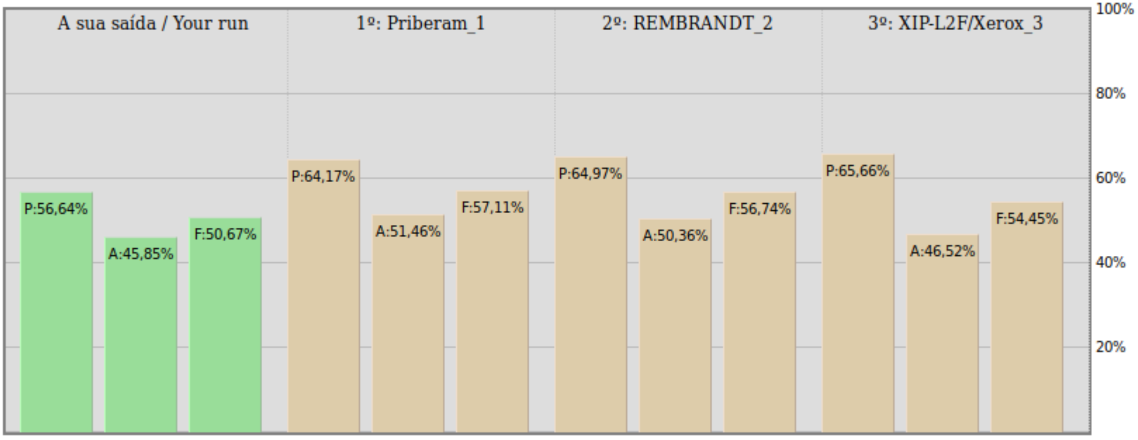
Figure 5.1: Results of Bidirectional LSTM-CNN in HAREM



Source: (CARDOSO, 2008b)

morphological class of the words internally, as part of process of pondering the influence of the context over each word, making the addition of POS-tagging data redundant.

Figure 5.2: Results of Bidirectional LSTM-CNN with POS-tagging in HAREM



Source: (CARDOSO, 2008b)

5.1 Discussion

The fact that the deep learning based system could not overcome the best HAREM participants in this case actually confirms a more general trend. Systems based on handcrafted rules succeed best in restricted linguistic environments while systems based on machine learning achieve medium quality results, but across a broader scope. Here, Chiu and Nichols (2015)'s system has no inner rules that attach it to a specific language, changing the word embeddings is enough to make it suitable to a new environment. Future work can be done on using the tag score raw value to choose multiple alternative tags and solve the problem of classification vagueness. The problem of identification vagueness, however, would require another kind of solution. Moreover, experimentation with BIOES tagging instead of BIO can be pursued in order to verify if there is any significant gain. Chiu and Nichols (2015) reported an improvement when using this scheme of tagging, but on Raj (2018)'s implementation BIO was enough to achieve a similar result as the original. A final suggestion is checking whether the addition of types and subtypes in the classification step would really degrade the performance of the system.

6 CONCLUSION

This work aimed to test whether applying a deep learning approach to the problem of named entity recognition in Portuguese would result in an advancement of the state-of-the-art. The method chosen was to adapt the work of Chiu and Nichols, originally made for English and tested on CoNLL, to the HAREM contest and compare its performance to the three best participants. Another aim was test whether equipping the new system with POS-tagging data would improve its F-score. We evaluated the system with SAHARA and discovered that, without POS-tagging, its F-score stayed behind the worst participant by 4.07%. When we added POS-tagging data with CoreNLP's module, the improvement seen was insignificant, only 0.29% on the F-score. Thus, the experiment shows that adding morphological information will not make this system better, and other means should be sought. It also shows that, for Portuguese, a deep learning based system can achieve results very close to the ones based on hand crafted rules, although the latter are still the best options. Further research should concentrate on better adapting the system to the HAREM environment.

REFERENCES

AMARAL, C. et al. Adaptação do sistema de reconhecimento de entidades mencionadas da Priberam ao HAREM. In: MOTA, C.; SANTOS, D. (Ed.). **Desafios na avaliação conjunta do reconhecimento de entidades mencionadas: O Segundo HAREM**. [S.l.: s.n.], 2008. p. 171–179.

AMARAL, D. F. do et al. O reconhecimento de entidades nomeadas por meio de conditional random fields para a língua portuguesa. Pontifícia Universidade Católica do Rio Grande do Sul, 2013.

CARDOSO, N. REMBRANDT - Reconhecimento de Entidades Mencionadas Baseado em Relações e ANálise Detalhada do Texto. In: MOTA, C.; SANTOS, D. (Ed.). **Desafios na avaliação conjunta do reconhecimento de entidades mencionadas: O Segundo HAREM**. [S.l.: s.n.], 2008. p. 195–211.

CARDOSO, N. SAHARA - Serviço de Avaliação HAREM Automático. In: MOTA, C.; SANTOS, D. (Ed.). **Desafios na avaliação conjunta do reconhecimento de entidades mencionadas: O Segundo HAREM**. [S.l.: s.n.], 2008. p. 347–354.

CARDOSO, N.; SANTOS, D. Directivas para a identificação e classificação semântica na colecção dourada do HAREM. In: SANTOS, D.; CARDOSO, N. (Ed.). **Reconhecimento de entidades mencionadas em português: Documentação e actas do HAREM, a primeira avaliação conjunta na área**. [S.l.: s.n.], 2007. p. 211–238.

CARVALHO, P. et al. Segundo HAREM: Modelo Geral, novidades e avaliação. In: MOTA, C.; SANTOS, D. (Ed.). **Desafios na avaliação conjunta do reconhecimento de entidades mencionadas: O Segundo HAREM**. [S.l.: s.n.], 2008. p. 11–31.

CHINCHOR, N.; ROBINSON, P. Appendix E: MUC-7 Named Entity Task Definition (version 3.5). In: **Seventh Message Understanding Conference (MUC-7): Proceedings of a Conference Held in Fairfax, Virginia, April 29 - May 1, 1998**. [s.n.], 1998. Available from Internet: <http://aclweb.org/anthology/M98-1028>.

CHIU, J. P. C.; NICHOLS, E. Named Entity Recognition with Bidirectional LSTM-CNNs. **ArXiv e-prints**, nov. 2015.

CHOLLET, F. et al. **Keras**. 2015. <https://keras.io>.

COLLOBERT, R.; WESTON, J. A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning. In: **Proceedings of the 25th International Conference on Machine Learning**. New York, NY, USA: ACM, 2008. (ICML ’08), p. 160–167. ISBN 978-1-60558-205-4. Available from Internet: <http://doi.acm.org/10.1145/1390156.1390177>.

CORTES, E. **Named-Entity-Recognition-with-Bidirectional-LSTM-CNNs**. [S.l.]: GitHub, 2018. <https://github.com/eduardogc8/egc-pyutils>.

DODDINGTON, G. et al. The Automatic Content Extraction (ACE) Program Tasks, Data, and Evaluation. In: **Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC-2004)**. Lisbon, Portugal: European Language Resources Association (ELRA), 2004. ACL Anthology Identifier: L04-1011. Available from Internet: <http://www.lrec-conf.org/proceedings/lrec2004/pdf/5.pdf>.

HAGèGE, C.; BAPTISTA, J.; MAMEDE, N. Reconhecimento de entidades mencionadas com o XIP: Uma colaboração entre a Xerox e o L2F do INESC-ID Lisboa. In: MOTA, C.; SANTOS, D. (Ed.). **Desafios na avaliação conjunta do reconhecimento de entidades mencionadas: O Segundo HAREM**. [S.l.: s.n.], 2008. p. 261–274.

HARTMANN, N. et al. Portuguese Word Embeddings: Evaluating on Word Analogies and Natural Language Tasks. **CoRR**, abs/1708.06025, 2017. Available from Internet: <http://arxiv.org/abs/1708.06025>.

KARPATHY, A. **CS231n: Convolutional Neural Networks Spring 2017**. [S.l.]: Stanford, 2017.

MANNING, C. D. et al. The Stanford CoreNLP natural language processing toolkit. In: **Association for Computational Linguistics (ACL) System Demonstrations**. [s.n.], 2014. p. 55–60. Available from Internet: <http://www.aclweb.org/anthology/P/P14/P14-5010>.

MERCHANT, R.; OKUROWSKI, M. E.; CHINCHOR, N. The multilingual entity task (met) overview. In: ASSOCIATION FOR COMPUTATIONAL LINGUISTICS. **Proceedings of a workshop on held at Vienna, Virginia: May 6-8, 1996**. [S.l.], 1996. p. 445–447.

MOTA, C. et al. É tempo de avaliar o TEMPO. In: MOTA, C.; SANTOS, D. (Ed.). **Desafios na avaliação conjunta do reconhecimento de entidades mencionados: O Segundo HAREM**. [S.l.: s.n.], 2008. p. 55–75.

MOTA, C. et al. Apresentação detalhada das colecções do Segundo HAREM. In: MOTA, C.; SANTOS, D. (Ed.). **Desafios na avaliação conjunta do reconhecimento de entidades mencionadas: O Segundo HAREM**. [S.l.: s.n.], 2008. p. 355–377.

PENNINGTON, J.; SOCHER, R.; MANNING, C. D. Glove: Global vectors for word representation. In: **EMNLP**. [S.l.: s.n.], 2014. v. 14, p. 1532–1543.

RAJ, K. **Named-Entity-Recognition-with-Bidirectional-LSTM-CNNs**. [S.l.]: GitHub, 2018. <https://github.com/kamalkraj/Named-Entity-Recognition-with-Bidirectional-LSTM-CNNs>.

SANG, E. F. T. K.; MEULDER, F. D. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In: DAELEMANS, W.; OSBORNE, M. (Ed.). **Proceedings of CoNLL-2003**. [S.l.]: Edmonton, Canada, 2003. p. 142–147.

SANTOS, D.; CARDOSO, N. Breve Introdução ao Harém. In: SANTOS, D.; CARDOSO, N. (Ed.). **Reconhecimento de entidades mencionadas em português: Documentação e actas do HAREM, a primeira avaliação conjunta na área**. [S.l.]: Linguateca, 2007. p. 1–16.

SANTOS, D. et al. Segundo HAREM: Directivas de anotação. In: MOTA, C.; SANTOS, D. (Ed.). **Desafios na avaliação conjunta do reconhecimento de entidades mencionadas: O Segundo HAREM**. [S.l.: s.n.], 2008. p. 277–286.

SCHUSTER, M.; PALIWAL, K. Bidirectional recurrent neural networks. **Trans. Sig. Proc.**, IEEE Press, Piscataway, NJ, USA, v. 45, n. 11, p. 2673–2681, nov. 1997. ISSN 1053-587X. Available from Internet: <http://dx.doi.org/10.1109/78.650093>.

SUNDHEIM, B. M. Overview of Results of the MUC-6 Evaluation. In: **Proceedings of the 6th Conference on Message Understanding**. Stroudsburg, PA, USA: Association for Computational Linguistics, 1995. (MUC6 '95), p. 13–31. ISBN 1-55860-402-2. Available from Internet: <https://doi.org/10.3115/1072399.1072402>.

TOUTANOVA, K. et al. Feature-rich part-of-speech tagging with a cyclic dependency network. In: **NAACL '03: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology**. Morristown, NJ, USA: Association for Computational Linguistics, 2003. p. 173–180. Available from Internet: <http://portal.acm.org/citation.cfm?id=1073445.1073478>.