Universidade Federal de Santa Catarina
Centro de Ciências, Tecnologias e Saúde
Coordenadoria Especial Interdisciplinar de
Tecnologias da Informação e Comunicação

Diefferson Koderer Môro

RECONHECIMENTO DE ENTIDADES NOMEADAS EM DOCUMENTOS DE LÍNGUA PORTUGUESA

Araranguá
2018

Diefferson Koderer Môro

RECONHECIMENTO DE ENTIDADES NOMEADAS EM DOCUMENTOS DE LÍNGUA PORTUGUESA

Trabalho de Conclusão de Curso apresentado à Universidade Federal de Santa Catarina como parte dos requisitos necessários para a obtenção do Título de Bacharel em Tecnologias da Informação e Comunicação.
Orientador: Prof. Dr. Vinicius Faria Culmant Ramos

Universidade Federal de Santa Catarina
Centro de Ciências, Tecnologias e Saúde
Coordenadoria Especial Interdisciplinar de
Tecnologias da Informação e Comunicação

Araranguá
2018

Ficha Catalográfica

Diefferson Koderer Môro
RECONHECIMENTO DE ENTIDADES NOMEADAS EM DOCUMENTOS DE LÍNGUA PORTUGUESA - Araranguá, 2018 - 37 p., 30 cm.
Orientador: Prof. Dr. Vinicius Faria Culmant Ramos
1. Reconhecimento de Entidades Nomeadas. 2. Língua Portuguesa. 3. Processamento de Linguagem Natural. 4. Aprendizado de Máquina
I. Universidade Federal de Santa Catarina. Tecnologias da Informação e Comunicação. II. RECONHECIMENTO DE ENTIDADES NOMEADAS EM DOCUMENTOS DE LÍNGUA PORTUGUESA.

Diefferson Koderer Môro

RECONHECIMENTO DE ENTIDADES NOMEADAS EM DOCUMENTOS DE LÍNGUA PORTUGUESA

Trabalho de Conclusão de Curso apresentado à Universidade Federal de Santa Catarina como requisito parcial para a obtenção do título de Bacharel em Tecnologias da Informação e Comunicação.

Patricia Jantsch Fiuza, Dra.
Coordenadora do Curso de Bacharelado em Tecnologias da Informação e Comunicação

Comissão Examinadora

Prof. Dr. Vinicius Faria Culmant Ramos
Universidade Federal de Santa Catarina
Orientador

Prof. Dr. Cristian Cechinel
Universidade Federal de Santa Catarina

Prof. Dr. Gustavo Medeiros de Araújo
Universidade Federal de Santa Catarina

Araranguá, 29 de novembro de 2018

Agradecimentos

Ao professor Vinicius, que me aturou e ensinou ao longo desses 3 anos, por tantos projetos trabalhados e por sempre acreditar no meu potencial.

Aos meus pais e irmã pelo carinho, apoio e incentivo, por sempre estarem ao meu lado e me ajudando a ser uma pessoa melhor.

Aos meus colegas e amigos, em especial para o Juary, Martinho, Lissandro pela amizade, apoio, maluquices e aprendizados, com toda certeza seria três anos "sem grança" sem vocês.

À Universidade Federal de Santa Catarina (UFSC) pelo acesso a informação, pela formação que recebi, pelos professores que sempre zelam pela excelência no ensino e pelas pessoas que conheci ao longo dos anos da minha graduação.

Aos professores Cristian e Gustavo por terem aceito participar da banca examinadora.

Resumo

Atualmente existe um grande volume de documentos pessoais e oficias, que trafegam na internet em diversos formatos, tais como doc, pdf, txt, que podem ter seus conteúdos analisados computacionalmente e assim agilizar em diversos processos executados com tais informações. Desta forma faz-se necessário a existência de procedimentos para realizar a análise destes documentos, e uma das ferramentas para esta tarefa é o Reconhecimento de Entidades Nomeadas (REN). Este trabalho tem como objetivo realizar um estudo sobre a aplicação e resultados que se pode obter em documentos redigidos de acordo com a norma culta da língua portuguesa. Para tal estudou-se os conceitos fundamentais relacionados ao Processamento de Linguagem Natural (PLN) e o tratamento de documentos em formato PDF. Foi feito um estudo de algumas ferramentas e corpus existentes, para textos escritos em português. Utilizou-se o *framework* spaCy, juntamente com o corpus HAREM e alguns documentos oficiais. A automatização na detecção de entidades nomeadas e seus vínculos em documentos escritos em língua portuguesa, pode ser viável utilizando-se as ferramentas e bases de dados já existentes. Entretanto, a dificuldade ainda são grandes e resultados que podem ser melhorados, visto que a identificação correta dessas entidades ainda não superou os 90% de acurácia.

Palavras-Chave: 1. Reconhecimento de Entidades Nomeadas. 2. Língua Portuguesa.

3. Processamento de Linguagem Natural. 4. Aprendizado de Máquina

Abstract

Currently there is a large volume of personal and official documents, which circulate in internet in several formats, such as doc, pdf, txt, which can have their contents analyzed computationally and thus streamline in several processes executed with such information. In this way it is necessary to have procedures to perform the analysis of these documents, and one of the tools for this task is the Named Entities Recognition (NER). This task aims to conduct a study on the application and results that can be obtained in documents drafted according to the cultured norm of the Portuguese language. For that, we studied the fundamental concepts related to the Natural Language Processing (NLP) and the treatment of documents in PDF format. There was a study of some existing tools and corporations, for texts written in Portuguese. We used the spaCy framework, along with the HAREM corpus and some official documents. The automation in the detection of named entities and their links in documents written in Portuguese language can be viable using existing tools and databases. However, the difficulty is still great and results can be improved, since the correct identification of these entities hasn't yet exceeded 90% accuracy.

Keywords: 1. Named Entities Recognition. 2. Portuguese Language. 3. Natural Language Processing. 4. Machine Learning.

Lista de figuras

Lista de tabelas

Lista de Siglas e Abreviaturas

CRF	*Conditional Random Fields*
EN	*Entidade Nomeada*
IA	*Inteligência Artificial*
PLN	*Processamento de Linguagem Natural*
POS	*Part of Speech*
REN	*Reconhecimento de Entidades Nomeadas*

Sumário

1 Introdução

A informatização tem sido uma constante e prevê-se que mantenha tal tendência. A informação nas últimas décadas, que antes era guardada em papel ou em outro meio físico, passou a estar no suporte digital. Fotografias, documentos, música, filmes e jogos são exemplos de informações armazenadas e distribuídas digitalmente [MORAIS, 2016].

Segundo Fred [2018], cerca de 90% dos dados existentes hoje em dia foram gerados nos últimos 2 anos. Em torno de 80% destes dados têm forma não estruturada ou semi-estruturada, sendo estas informações vindas de e-mails, publicações de blog, redes sociais, textos gerados em sistemas de suporte ao cliente, sensores, imagens digitais, vídeos, etc [FRED, 2018]. Nesse caso, um usuário do sistema que utiliza esse tipo de dado é obrigado a interpretar os dados para poder compreender as informações contidas neles. Segundo um estudo da DOMO [2018] estima-se que até 2020 cada pessoa no planeta esteja gerando 1,7 MB (MegaBytes) de dados por segundo.

Com essa grande quantidade de informações disponibilizadas na web, é necessário a existência de métodos para realizar a filtragem dos dados relevantes e apresentá-los aos leitores. O Processamento de Linguagem Natural (PLN) é um campo da Inteligência Artificial (IA) voltado para tornar a linguagem humana compreensível para os computadores. Isso permite obter informações estruturadas de forma que elas possam ser indexadas e usadas por uma máquina para tarefas orientadas por conhecimento, como resposta a perguntas [PIRES, 2017]. Dados em formato de texto são dados desestruturados que, normalmente, pertencem a uma linguagem específica que possui uma sintaxe e uma semântica associada. Qualquer extrato de texto, como documentos, palavras ou frases, estão relacionados diretamente às linguagens naturais na maioria das vezes, portanto, uma linguagem "natural" é uma linguagem usada por seres humanos de forma natural para se comunicar ao invés de construída ou criada artificialmente, como são as linguagens de programação de computadores.

Os idiomas português, inglês ou chinês são exemplos de linguagens naturais. Elas são utilizadas para que humanos possam se manifestar de diferentes maneiras, como numa comunicação oral, um discurso, por escrito ou até por sinais. Apesar da estrutura sintática e gramatical existentes nas linguagens naturais, o processamento e análise de textos são um grande desafio computacional, especialmente para a IA. Isso porque essas linguagens são especialmente diferentes das linguagens de programação computacionais, como é possível observar nos códigos 1.1 e 1.2, pois estas são estruturadas e possuem sintaxe e padrões regulares, portanto, é muito difícil aplicarmos, diretamente, modelos matemáticos ou estatísticos diretamente às linguagens naturais.

```
1 Sabe-se que o somatório de dois números quaisquer resulta em um novo valor,
      como na demonstração seguinte:
```

```
A soma de 2 mais 2 possui como resultado o valor 4
Sendo assim a afirmação abaixo é verdadeira.
1 + 1 + 1 + 1 é igual a 4
```

Código–fonte 1.1 – Linguagem Natural

```
n1 = 2
n2 = 2

print('Sabe-se que o somatorio de dois numeros quaisquer resulta em um novo valor, como na demonstracao seguinte:')
res = n1 + n2
print('A soma de {} mais {} possui como resultado o valor {}'.format(n1, n2, res))
print('Sendo assim a afirmacao abaixo eh verdadeira.')
sum = 0
for i in range(res):
    sum += 1
print(sum == res)
```

Código–fonte 1.2 – Linguagem de Programação

O objetivo do PLN é fornecer aos computadores a capacidade de entender, compor, reconhecer o contexto, fazer análise sintática, semântica, léxica e morfológica, criar resumos, extrair informação, interpretar os sentidos, analisar sentimentos e até aprender conceitos com os textos processados.

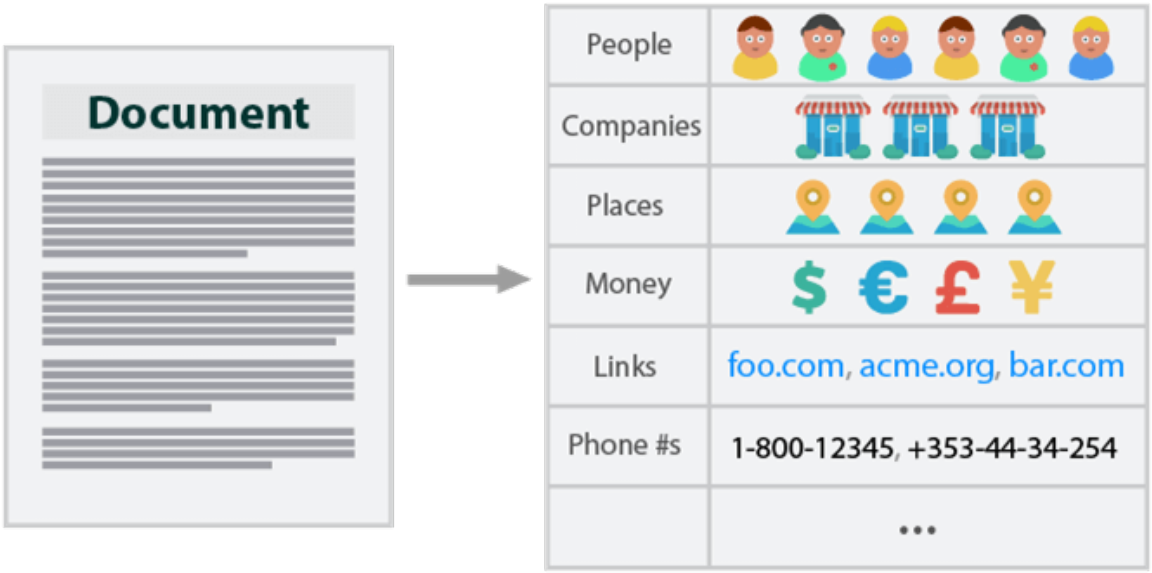
Desta forma existem vários desafios para que os computadores possam realizar o "entendimento" dos textos em linguagens naturais, um deles é o Reconhecimento de Entidades Nomeadas (REN). Este consiste na identificação de palavras, conforme a ilustração 1, que é a menor unidade em uma linguagem, sendo estas independentes e com significados próprios, que se encontram em textos de forma livre.

Figura 1 – Identificação de palavras

Fonte – Fred [2018]

O REN é parte do processo de identificação das Partes do Discurso (*Part of Speech* - POS) que, em sua maioria, são nomes próprios, verbos, adjetivos ou advérbios. A partir dessa identificação, o objetivo do REN é classificá-la dentro de um conjunto de categorias pré-definidas (figura 2), tais como Pessoa, Organização ou Local, as quais remetem a uma referência específica [FONSECA; CHIELE; VANIN, 2015].

Figura 2 – Conjunto de categorias



Fonte – Fred [2018]

O REN é uma tarefa fundamental do PLN por duas razões principais: o primeiro é

que o REN é usado em muitas áreas de pesquisa, por exemplo, genes e proteínas podem ser considerados entidades nomeadas e muitos trabalhos na medicina focam na análise científica de artigos a fim de encontrar as relações entre os mesmos e conduzir uma pesquisa experimental [ATDAĞ; LABATUT, 2013]. Em segundo lugar, ele é usado por ferramentas avançadas de PLN para encontrar relações ou extrair informações de textos.

1.1 Delimitação

Extrair informações semânticas de um documento não é uma tarefa trivial para os computadores, exigindo assim uma área de estudos dedicada. Um documento pode estar disponível nos mais variados formatos, como texto plano (DOCX, ODT, TXT), PDF, imagem, bem como estruturado em formas diferentes. Então, propor um estudo que englobe todos os tipos e formatos de documentos existentes é consideravelmente inviável.

Para este trabalho estudou-se apenas documentos PDFs por ser um formato difundido mundialmente e com uma grande utilização por parte de órgãos públicos, limitando o universo de documentos estudados. Esperando-se que desta forma seja possível aproveitar as características intrínsecas a esse domínio na extração de elementos semânticos.

1.2 Objetivos

A seguir serão descritos os objetivos geral e específicos adotados.

1.2.1 Objetivo Geral

Realizar estudos sobre o reconhecimento de entidades nomeadas em documentos redigidos de acordo com a norma culta da língua portuguesa.

1.2.2 Objetivos Específicos

- Identificar algoritmos e técnicas usados na literatura sobre o reconhecimento de entidades nomeadas aplicados à língua portuguesa e outras línguas latinas.
- Obter e/ou construir um Corpo de Texto (Text Corpora) com anotações e utilidades para servir de exemplo para o treinamento dos algoritmos e técnicas de REN.
- Comparar os algoritmos e técnicas usadas para a língua inglesa, consolidada, para o REN em documentos da língua portuguesa.

2 Referencial teórico

2.1 Processamento de linguagem natural

Consiste no desenvolvimento de modelos computacionais para a realização de tarefas que dependem de informações expressas em alguma língua natural (e.g. tradução e interpretação de textos, busca de informações em documentos e interface homem-máquina) [PEREIRA, 2016].

Conforme Pereira [2016], a pesquisa em PLN está voltada, essencialmente, a três aspectos da comunicação em língua natural:

- *som*: fonologia
- *estrutura*: morfologia e sintaxe
- *significado*: semântica e pragmática

Pereira [2016] diz que a:

> *fonologia* está relacionada ao reconhecimento dos sons que compõem as palavras de uma língua. A *morfologia* reconhece as palavras em termos das unidades primitivas que a compõem (*e.g. caçou* → *caç+ou*). A *sintaxe* define a estrutura de uma frase, com base na forma como as palavras se relacionam nessa frase (Figura 3). A *semântica* associa significado a uma estrutura sintática, em termos dos significados das palavras que a compõem (*e.g* à estrutura da Figura 3, podemos associar o significado "*um animal perseguiu/capturou outro animal*"). Finalmente, a *pragmática* verifica se o significado associado à uma estrutura sintática é realmente o significado mais apropriado no contexto considerado (*e.g.* no contexto predador-presa, "*perseguiu/capturou*" → "*comeu*").

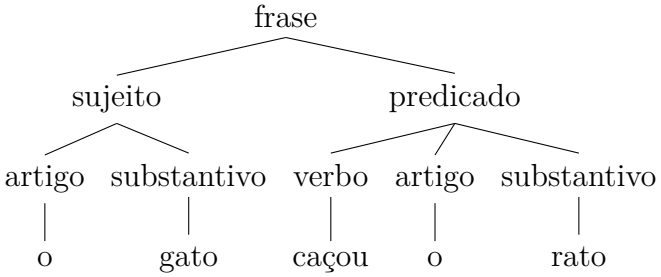


Figura 3 – Árvore sintática

Fonte – Adaptado de Pereira [2016]

2.1.1 Extração da Informação

Uma das tarefas do PLN, a Extração da Informação (EI) é parte fundamental para a realização do REN, compreendendo a tarefa de extrair informações estruturadas de docu-

mentos legíveis para máquinas organizados de forma desestruturada e/ou semi-estruturada [ADANIYA; JR, 2009].

De acordo com Amaral et al. [2017]:

> Sistemas comuns de EI caracterizam-se por três etapas: análise do texto, seleção de regras e aplicação de regras. A primeira etapa realiza desde a segmentação do texto em sentenças até uma análise linguística mais completa como a identificação de palavras-chave como nomes próprios, verbos de interação e a relação de sucessor e antecessor das palavras que serão extraídas dos textos. A segunda define as regras de extração de informação, que são associadas a "triggers", geralmente, palavras-chave. A presença de "triggers" ativa a verificação de partes condicionais de regras de correspondência. Por exemplo, uma determinada regra pode estar associada a ocorrência de prefixos e/ou sufixos nas palavras relevantes num conjunto de textos. A terceira e última etapa exprime que ao acionar uma regra, todas as suas condições contextuais são verificadas, e um modelo é preenchido de acordo com os resultados das regras apropriadas para um determinado domínio. Dessa forma, o resultado pode ser a geração de um modelo ou de um texto anotado. (p. 19).

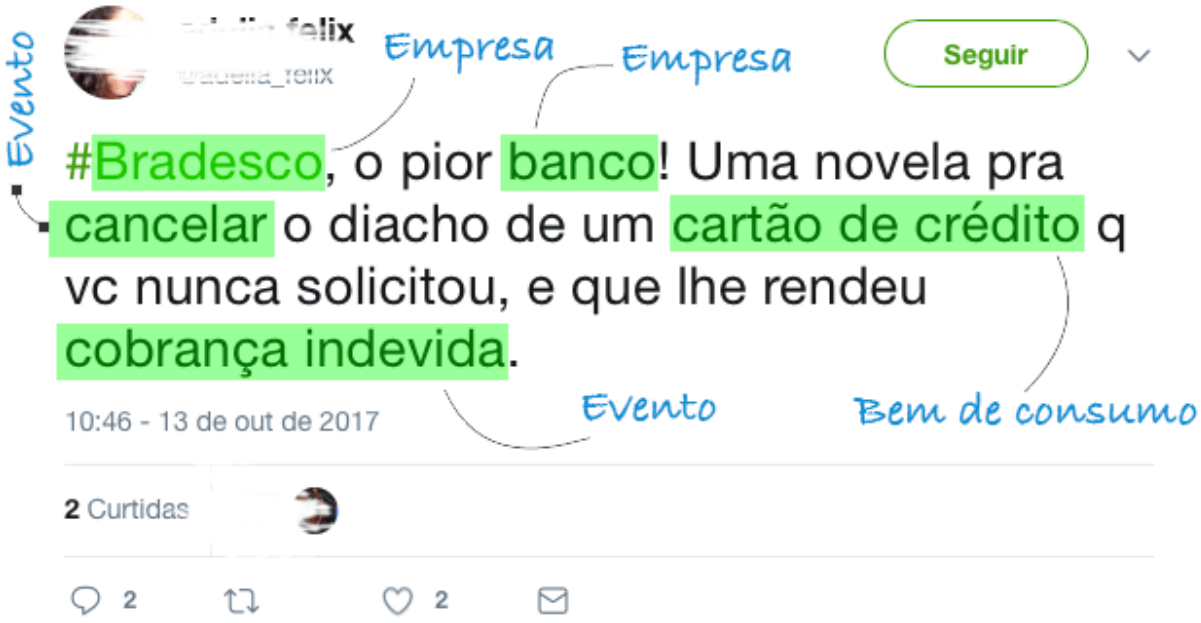
Deste modo, a EI tem como passo fundamental identificar, classificar e relacionar Entidades Nomeadas (EN), processo denominado como Reconhecimento de Entidades Nomeadas. O qual, busca a identificação de expressões linguísticas contidas em textos que possam ser associadas a determinadas classes ou categorias.

2.1.1.1 Reconhecimento de Entidades Nomeadas

O REN é uma sub-tarefa da EI que permite localizar e classificar Entidades Nomeadas mencionadas em textos desestruturados em categorias pré-definidas como pessoas, lugares, organizações ou valores monetários. Há, ainda, as classificações mais específicas, que estão de acordo com o domínio.

Na figura 4 temos um exemplo de tweet, nele podemos encontrar algumas EN do tipo empresa, evento e bem de consumo, realçadas com a cor verde, essa demarcação ou realce é conhecido como anotação, e a classificação de cada anotação é chamado de entidade. As classes que serão reconhecidas depende do que se deseja analisar, supomos que o Bradesco que é uma instituição bancária, ou seja pessoa jurídica, quer saber o que as pessoas estão "falando" sobre ela, através do reconhecimento de entidades nomeadas na rede social do Twitter, por exemplo, ela poderia analisar as mensagens das pessoas e levantar as informações como deste tweet de exemplo, assim focariam a análise para entidades do tipo empresa e que demonstre eventos e sentimentos, nesse caso, figura 4, eles teriam como resultado a insatisfação do cliente dizendo que para cancelar o cartão de crédito é uma novela, ou seja, é difícil de conseguir. Esse tipo de abordagem, já está bem difundida e já possui até nome, conhecido como análise de sentimentos.

Figura 4 – Identificação de entidades



Fonte – Fred [2018]

2.1.2 Corpus

Segundo Carvalho [2012] o corpus é uma coleção especial de textos coletados conforme critérios específicos, e é um dos principais requisitos para o processamento estatístico da linguagem natural. Tal recurso é utilizado para treinamento e teste de modelos estatísticos de linguagem natural escrita e falada, bem como para avaliação de componentes de sistemas de linguagem natural. Abaixo na figura 5 temos um exemplo, através de um trecho do corpus do segundo HAREM.

Figura 5 – Trecho corpus segundo HAREM

```
▼<colHAREM versao="Segundo_dourada_com_relacoes_14Abril2010">
 ▼<DOC DOCID="H2-dftre765">
    <P>Fatores Demográficos e Econômicos Subjacentes</P>
   ▼<P>
     A revolta histórica produz normalmente uma nova forma de pensamento quanto à forma de organização da
     sociedade. Assim foi com a
     <EM ID="H2-dftre765-1" CATEG="ABSTRACCAO|ACONTECIMENTO" TIPO="IDEIA|EFEMERIDE">Reforma Protestante</EM>
     . No seguimento do colapso de instituições monásticas e do escolasticismo nos finais da
     <EM ID="H2-dftre765-102" CATEG="OUTRO" COMENT="DUVIDA_DIRECTIVASTEMPO">Idade Média</EM>
     na
     <EM ID="H2-dftre765-37" CATEG="LOCAL" TIPO="HUMANO" SUBTIPO="DIVISAO">Europa</EM>
     , acentuado pela "
    ▼<OMITIDO>
       <EM ID="H2-dftre765-17" CATEG="ACONTECIMENTO" TIPO="EFEMERIDE">Cativeiro Babilónica da igreja</EM>
     </OMITIDO>
     " no papado de
     <EM ID="H2-dftre765-11" CATEG="ACONTECIMENTO" TIPO="EVENTO">Avignon</EM>
     , o
     <EM ID="H2-dftre765-6" CATEG="ACONTECIMENTO|ABSTRACCAO" TIPO="EFEMERIDE|IDEIA" COREL="H2-dftre765-102"
     TIPOREL="ACONTECIMENTO**outrarel**H2-dftre765-102**OUTRO">Grande Cisma</EM>
     e o fracasso da conciliação, assistimos
     <EM ID="H2-dftre765-72" CATEG="TEMPO" TIPO="TEMPO_CALEND" SUBTIPO="DATA">no século XVI</EM>
     ao fermentar de um enorme debate sobre a reforma da religião e dos posteriores valores religiosos
     fundamentais. Este debate passou completamente ao lado de
     <EM ID="H2-dftre765-8" CATEG="PESSOA" TIPO="POVO">Portugal</EM>
     , demasiado distante do foco onde surgiram estes pensamentos. A imprensa, inventada na
     <EM ID="H2-dftre765-9" CATEG="LOCAL" TIPO="HUMANO" SUBTIPO="PAIS" COREL="H2-dftre765-37"
     TIPOREL="incluido">Alemanha</EM>
```

Fonte – Elaborado pelo autor

2.1.2.1 Texto Marcado

Dos corpus de textos puros (textos sem marcações) pode ser extraído bastante conhecimento e por isso tais corpus são muito úteis. Várias técnicas fazem uso de um tipo especial de corpus, conhecido como texto marcado ou corpus anotado. Este é criado através de um processo conhecido como anotação, onde informações estruturais são inseridas no texto [CARVALHO, 2012]. Em alguns deles, apenas as estruturas básicas são marcadas, tais como as fronteiras de sentenças e parágrafos, conforme a seção 2.1.3.2. Já outros possuem uma carga de informação maior tal como toda a estrutura sintática. A marcação mais comum é a codificação das categorias das palavras [CARVALHO, 2012].

2.1.3 Técnicas de PLN

Nesta seção serão elicitadas três técnicas necessárias para o Processamento de Linguagem Natural.

2.1.3.1 Tokenização

Geralmente um dos primeiros passos no PLN é segmentar o texto recebido em unidades chamadas *tokens*. Onde cada *token* representa uma palavra, número ou sinal de pontuação [CARVALHO, 2012].

> Em termos de classificação, uma determinada palavra pode ser inserida numa de duas classes: *open word* e *close word* definindo assim se o seu significado é mutável ou não. Por outro lado, através da verificação do

> seu conteúdo semântico podem ser classificadas como *lexical words* (nomes, verbos, adjectivos ou advérbios) e *functional words* (artigos, pronomes, conjunções e preposições). Apesar de classes independentes, estas estão ligadas pois a maior parte das *functional words* pertencem a classe das *closed words*, enquanto que a maioria das *lexical words* pertencem à classe das *open words* [PEREIRA, 2014, p. 11].

2.1.3.2 Separação de Sentenças

Sentenças podem ser consideradas como sendo uma sequência de palavras ou caracteres que está entre os delimitadores "?", "." ou "!". Além de indicar o fim da sentença, o sinal de pontuação pode também indicar abreviação, ou ambas as funções simultaneamente, que neste último caso é um fenômeno chamado haplologia [CARVALHO, 2012]. Ainda segundo Carvalho [2012], existem frases que contêm outras frases dentro delas, a exemplo das ocorrências de frases com "" e ().

2.1.3.3 Análise Morfossintática

No PLN a tarefa que identifica a classe gramatical de cada uma das palavras de uma sentença é conhecida como Análise Morfossintática. Geralmente tais classes são representadas por um conjunto de códigos, e estes são utilizados na etiquetagem das palavras [CARVALHO, 2012].

Carvalho [2012] argumenta que:

> Uma das dificuldades desta tarefa é a existência de muitas palavras com diferentes classificações possíveis. Tais palavras, se estiverem fora de contexto, ocasionam a ambiguidade sobre sua interpretação para a correta classificação gramatical. Por exemplo, na frase "Vamos assistir ao jogo", a palavra "jogo" é um substantivo que pode significar, dentre outras, uma partida de futebol. Porém, a mesma palavra empregada na frase "Eu jogo videogame", trata-se de uma flexão na primeira pessoa do singular do presente do indicativo do verbo "jogar". A palavra "jogo" é um exemplo de ambiguidade existente na língua portuguesa.(p. 6).

2.1.4 *Frameworks* de PLN

Para a execução deste trabalho foram pesquisados alguns *frameworks*, tais como NLTK, Apache OpenNLP, NERP-CRF e spaCy. A seguir será feita uma breve descrição dessas ferramentas.

NLTK (Natural Language Toolkit) fornece uma biblioteca de código aberto em Python de módulos para PLN, que contém vários recursos linguísticos. Dentre os algoritmos existentes na ferramenta, os principais são de tokenização de palavras e frases, classificadores e partes do discurso.

Apache OpenNLP apresenta-se como uma framework que fornece mecanismos para suportar funções de aprendizado de máquina. Esta framework reune em uma biblioteca de Java as tarefas mais típicas de PLN.

NERP-CRF tem como características a utilização da linguagem Python, licença de código aberto, utiliza aprendizado de máquina e foi desenvolvida na Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS). O NERP-CRF é um recurso de REN para a língua portuguesa, que reconhece e classifica 10 categorias de entidades nomeadas (Pessoa, Local, Organização, Obra, Abstração, Tempo, Coisa, Outro, Valor, Acontecimento). Como seu próprio nome sugere, a ferramenta utiliza o algoritmo de *Condicional Random Field* (CRF) e foi treinada por meio do corpus do HAREM.

Spacy é uma ferramenta para PLN de código aberto e escrito na linguagem Python, suporta mais de 33 línguas, possuindo 18 categorias de EN, além disso permite criar outras categorias, utiliza aprendizado de máquina e modelos de redes neurais convolucionais.

2.2 Avaliação de Sistemas de Reconhecimento de Entidades Nomeadas

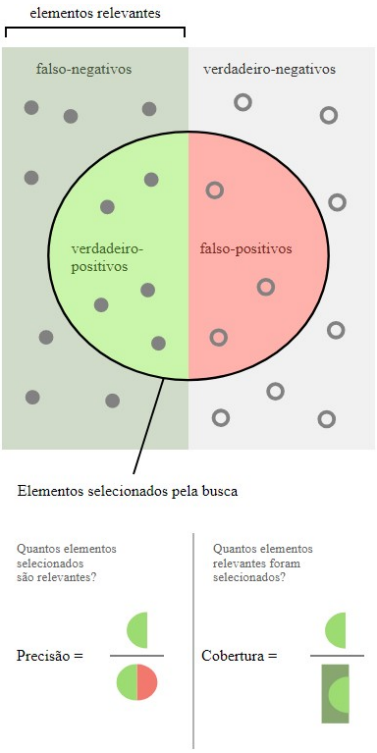
Conferências possuem uma grande importância para diversas áreas do conhecimento, e no REN isso não é diferente. Estas por sua vez, propuseram algumas técnicas para avaliar os sistemas de REN, elas consideraram a capacidade do sistema em anotar os textos tal qual faria um especialista linguista, comparando as saídas dos sistemas com textos anotados por humanos. Neste capítulo, veremos algumas medidas geralmente empregadas e algumas das conferências que foram importantes para a avaliação conjunta de sistemas REN.

2.2.1 Medidas de Avaliação

As avaliações de sistemas de REN são geralmente baseadas na comparação das saídas dos sistemas com textos anotados por especialistas.

As medidas de avaliação levam em conta noções de precisão (*precision*), cobertura (*recall*), sobre-geração (*overgeneration*) e medida-F (*F-measure*), que são métricas empregadas nas avaliações de sistemas de *Recuperação da Informação*. No contexto de REN, um item relevante é uma EN corretamente identificada e classificada por um sistema, porém há casos em que o sistema pode ser avaliado em apenas uma destas tarefas. Na figura 6 é possível visualizar informações mais detalhadas sobre precisão e cobertura.

Figura 6 – Precisão e Cobertura



Fonte – Modificado pelo autor, original de Observatório... [2018]

De forma geral, a precisão (P) define quanto da informação extraída é correta. A cobertura (C), quanta informação foi extraída. A sobre-geração é o quanto de informação extraída é supérflua. Quando a Cobertura aumenta, a Precisão tende a diminuir e vice-versa, pois são inversamente proporcionais. Precisão e Cobertura estão sempre no intervalo de [0; 1], sendo 0 o pior resultado e 1 o melhor.

$$F\beta = \frac{(\beta^2 + 1) \times P \times C}{\beta^2 \times P + C} \tag{2.1}$$

O parâmetro β da Equação 2.1 diferencia a importância da precisão e cobertura, e pode ser manipulado de acordo com a necessidade do sistema. A precisão e a cobertura ficam igualmente balanceadas quando atribuímos o valor `1` ao parâmetro β.

Assim, para a avaliação do sistema no uso com os documentos oficias a medida mais importante é a precisão, visto que devido a importância da correta marcação das entidades é preferível no lugar de possuir um número maior de entidades que possam induzir a erros.

2.2.2 Conferências de Avaliação de REN

2.2.2.1 MUC

O MUC (*Message Understanding Conferences*), foi uma série de eventos que ocorreram entre 1987 e 1998, e tinha como propósito avaliar e promover o progresso nas pesquisas

sobre extração de informações e padronizar a avaliação das tarefas dessa área.

Em 1995, ocorreu a sexta edição do MUC, onde teve início a avaliação do REN para a língua inglesa. Esta edição teve a sua peculiaridade em relação a outras edições, pois, as edições anteriores consideravam o REN como sendo uma parte da tarefa de Extração da Informação. Para o MUC a tarefa de REN tinha como objetivo reconhecer nome de pessoas, organizações, nome de lugares, expressões temporais e certos tipos de expressões numéricas.

2.2.2.2 CoNLL

A CoNLL (*Conference on Computational Natural Language Learning*) é uma série de conferências que tem como objetivo promover a pesquisa e avaliação em diversas áreas de PLN. Sua primeira edição data de 1997.

A conferência de 2002, CoNLL-2002, tinha como tarefa compartilhada a avaliação de sistemas de REN independente de linguagem. Neste evento foram considerados quatro tipos de EN: pessoas, lugares, organizações e nomes de entidades diversas que não se encaixam em nenhuma das categorias anteriores. Os participantes recebiam dados de treino e teste para duas línguas, holandesa e espanhola, e poderiam também utilizar recursos extras para treinamento, além dos dados fornecidos. Na conferência de 2003, CoNLL-2003, novamente o REN independente de linguagem foi foco da tarefa compartilhada, porém as línguas alvos desse evento foram as línguas inglesa e alemã. Uma das principais tarefas dos participantes deste segundo evento era descobrir como melhorar o desempenho dos seus sistemas de REN utilizando alguns recursos adicionais fornecidos pelo evento, tais como listas de EN e textos não-anotados.

2.2.2.3 HAREM

A Linguateca é um centro de recursos (distribuídos) para o processamento computacional da língua portuguesa. Tem como objetivo facilitar o acesso aos recursos já existentes tais como corpus, enciclopédias, textos em português; desenvolver em colaboração com os interessados, os recursos mais prementes além de organizar avaliações conjuntas que envolvam a comunidade científica de interesse em torno do PLN.

O HAREM (Avaliação de Sistemas de Reconhecimento de Entidades Mencionadas) é uma avaliação conjunta de sistemas de Reconhecimento de Entidades Mencionadas organizada pelo Linguateca, iniciado em 2005. Segundo Carvalho [2012], o HAREM é a primeira avaliação (conjunta) de sistemas de REN em português. Teve como motivação o fato de sentirem que os eventos de avalização de REN anteriores não tinham abordado a tarefa com profundidade suficiente, e o objetivo de reunir a comunidade científica em torno de outro evento de avaliação dentro do processamento da língua portuguesa. Nesse intuito,

os participantes tiveram um papel ativo na organização do HAREM, tendo colaborado na criação das diretivas e na anotação das coleções.

A avaliação do HAREM segue um conjunto de diretivas estabelecidas junto com os participantes do próprio HAREM, a qual denominaram *Diretivas de Avaliação do HAREM*. Essas diretivas representam o conjunto de pontuações, regras e medidas usadas para comparar as saídas dos sistemas em relação à *Coleção Dourada*, que é o texto de comparação utilizado pelo evento, criado em conjunto com a comunidade.

Até a realização deste trabalho o HAREM já organizou três eventos de avaliação conjunta:

- **Primeiro HAREM**: Evento organizado a partir de Setembro de 2004 com a realização da avalização conjunta em Fevereiro de 2005;
- **Mini HAREM**: Realizado em Abril de 2006, utilizou a mesma metodologia e a mesma plataforma de avaliação da primeira avaliação;
- **Segundo HAREM**: Organizado a partir de Novembro de 2007, teve a avaliação conjunta realizada em Abril de 2008.

3 Trabalhos relacionados

A identificação das relações entre entidades é um processo composto por diversos passos, começando pelo reconhecimento e localização das entidades nomeadas envolvidas, recolhimento e processamento dos contextos dessas ocorrências e a identificação dos grupos formados no relacionamento das entidades.

Para tal existem soluções relacionadas com cada um dos passos, onde as principais características de soluções são descritas a seguir. Amaral e Vieira [2014] citam que existe três principais abordagens para a extração de ENs, sendo elas os sistemas baseados em regras, em aprendizado de máquina e abordagens híbridas.

Segundo Fonseca, Chiele e Vanin [2015] existe atualmente uma quantidade razoável de ferramentas de REN disponíveis para o português, como NERP-CRF [1], FreeLing [2], Palavras [3] e OpenNLP [4]. Existe uma outra ferramenta chamada spaCy [5] que segundo Choi, Tetreault e Stent [2015] possui o analisador sintático mais rápido do mundo. Apesar de ele fazer o REN para a língua portuguesa, este é limitado, pois foi treinado somente com quatro categorias: pessoa, localização, organização e misto, podendo treinar o mesmo para outras. Para treinamento destas ferramentas normalmente se utiliza o Wikipédia, mas para o português existem coletâneas de documentos como o HAREM e Amazônia [FONSECA; CHIELE; VANIN, 2015] e o bosque [CONTIER; PADOVANI; NETO, 2012].

No trabalho de Fonseca, Chiele e Vanin [2015] é descrito que os corporas Amazônia e o HAREM tiveram papel fundamental, e optou-se por utilizar o OpenNLP juntamente com suas bibliotecas e classes de treino. Ressaltando que o mesmo argumenta que o seu trabalho difere da maioria por propor a utilização do corpus Amazônia para treinar o modelo e o corpus do segundo HAREM para avaliá-lo. No processo de avaliação o modelo gerado no trabalho de Fonseca, Chiele e Vanin [2015] demonstra percentuais totais de aproximadamente 38% para as 3 medidas mais utilizadas (precisão, cobertura e medida-f), onde as classes semânticas “Pessoa”, “Local” e “Organização” obtiveram os melhores resultados, atingindo respectivamente uma medida-F de 57.61%, 54.10% e 40.50%.

Já no trabalho de Carvalho [2012] optou-se por desenvolver um sistema de REN com aprendizado de máquina e realizar o comparativo com os resultados demonstrados pelos sistemas que participaram do Segundo HAREM, onde o mesmo cita que os resultados obtidos podem ser considerados bons considerando o número de sistemas participantes e o fato de que a maioria dos sistemas serem baseados em regras manuais. Carvalho

[1] <https://www.inf.pucrs.br/linatural/wordpress/index.php/recursos-e-ferramentas/nerp-crf/>
[2] <http://nlp.lsi.upc.edu/freeling/>
[3] <http://visl.sdu.dk/constraint_grammar.html>
[4] <http://opennlp.apache.org/>
[5] <https://spacy.io/>

[2012] demonstra que atingiu uma medida-F de 42,48%, logo é notável que obteve-se um resultado superior ao apresentado por Fonseca, Chiele e Vanin [2015].

Amaral et al. [2013] realizaram o desenvolvimento de um sistema para Reconhecimento de Entidades Nomeadas utilizando um método probabilístico de predição estruturada conhecido com *Conditional Random Fields* (CRF) e avaliou o desempenho com base no corpus do HAREM. Em seu trabalho Amaral et al. [2013] realizaram três testes, tendo como o "teste 2" o de melhor performance, cabe ressaltar o uso do estilo de notação chamado BILOU, levantado pela mesma como sendo o de melhor resultado se comparado ao BIO. Amaral et al. [2013] apresenta que através da comparação do seu trabalho com os sistemas que participaram da Conferência do Segundo HAREM, o mesmo seria o sistema com os melhores resultados para as medidas de precisão e medida-F.

Pires [2017] no seu trabalho realizou uma análise da performace base de algumas ferramentas selecionadas (Stanford CoreNLP, OpenNLP, spaCy e NLTK) com a coleção HAREM. Em seguida, ele efetuou um estudo aos hiperparâmetros, de modo a selecionar a melhor configuração para cada ferramenta, conseguindo alcançar melhorias em cada ferramenta. Pires [2017] também preparou um corpus Português único, denominado de SIGARRA News Corpus, composto por 905 notícias anotadas, com 12644 anotações de entidades.

4 Metodologia

Para atingir os nossos objetivos, optamos por fazer uma revisão sistemática sobre o Processamento de Linguagem Natural (PLN) e o Reconhecimento de Entidades Nomeadas (REN) para línguas latinas como o Português, Italiano e o Espanhol

Após a revisão da literatura, entendemos como necessário a realização de uma busca, extração, formatação e análise de bases de dados a serem utilizadas no treinamento dos sistemas. Esta etapa é fundamental para o desenvolvimento deste trabalho. Encontramos alguns corpus de texto com anotações para a língua portuguesa, mas não encontramos muitos trabalhos que asseguram a qualidade do corpus de texto. Portanto, acreditamos que uma das etapas deve ser o estudo desses corpora de texto para que possamos aplicar com segurança os algoritmos e técnicas para o REN em língua portuguesa. Para atingirmos o objetivo de termos um corpus de texto com anotações que possam nos auxiliar no REN, é necessário considerar os exemplos de anotações já existentes nesses corpora, pois a maior parte dos algoritmos e técnicas que as usam necessitam de: 1. uma quantidade considerável de exemplos, e 2. frases a serem analisadas com respectivas classificações (anotações) das entidades e classes gramaticais contidas na mesma.

Deste modo utilizaremos duas fontes, o corpus HAREM e os documentos oficiais, que por sua vez precisam de alguns tratamentos para tornar possível a sua utilização.

A próxima etapa consiste em preparar os dados coletados através da marcação das entidades para que estes possam ser utilizados nas etapas de treinamento e avaliação dos modelos. Os modelos são criados na etapa de treinamento, onde se define as bases (dados marcados) e o número de iterações a ser utilizado em cada treino.

Na última etapa, propomos a avaliação dos modelos através da medição das métricas descritas na seção 2.2.1.

5 Resultados

Abordamos a primeira etapa do nosso processo de revisão da literatura com o objetivo principal de identificação das principais ferramentas, técnicas e algoritmos utilizados para o reconhecimento de entidades nomeadas em textos em português.

Identificamos na literatura 4 principais ferramentas de análise de textos que fazem o REN para o português, são elas: NERP-CRF, OpenNLP, spaCy e NLTK. Além disso, identificamos, também, na literatura as principais bases de dados anotadas em português, que são: HAREM [1], BOSQUE [2], FLORESTA [3] e AMAZÔNIA [4].

Coletamos alguns documentos oficiais distribuídos em formato PDF, dentre estes alguns possuíam marca d'água, com isto fez-se necessário realizar estudos a cerca da extração de conteúdos para documentos PDF, enfatizando as possibilidades para a linguagem de programação Python, visto que para os *frameworks* de REN o texto de entrada precisa ser em formato plano, deste modo foram encontradas diversas bibliotecas para realizar o tratamento, destes escolhemos cinco, sendo elas o PDFMiner, pdftotext, pypdf2, textract

Utilizamos um documento para verificar os procedimentos necessários para realizar a extração, chegamos a resultados distintos para cada biblioteca, onde o PDFMiner obteve o melhor resultado, mas ainda não satisfatórios, visto que o mesmo não consegue extrair as tabelas de forma que possam ser aproveitadas e também havia espaços em excesso, sobre linhas, parágrafos separados, entre outros. Sendo assim, trabalhou-se com a implantação de heurísticas para tratar o resultado da extração, a fim de conseguir um texto com uma legibilidade melhor. O código responsável para tal feito encontra-se no Anexo A.1.

Com esta primeira etapa finalizada, seguimos para a escolha da ferramenta de análise que seria utilizada, diante das citadas no início desta sessão, optou-se pelo uso do spaCy visto ser, segundo a literatura uma boa ferramenta de REN.

Em seguida realizou-se a marcação das entidades para utilizar nos treinos e avaliações. Como testes iniciais foram realizados alguns treinos e validações, resultados na tabela 1.

Neste primeiro momento, o nosso conjunto de modelos ficou:

- Modelo 1 - modelo original disponibilizado no SpaCy, versão utilizada 2.0.0;
- Modelo 2 - modelo gerado com base no original, acrescido do treino de 500 iterações dos "dados de treino" e 20 iterações nos dados dos arquivos de Localidades;
- Modelo 3 - modelo gerado com base no original, acrescido do treino de 20 iterações dos "dados de treino";

[1] <http://www.linguateca.pt/HAREM/PacoteRecursosSegundoHAREM.zip>
[2] <https://www.linguateca.pt/Floresta/ficheiros/Bosque_CP_7.5_cgde_22032016.conll.gz>
[3] <https://www.linguateca.pt/Floresta/ficheiros/FlorestaVirgem_CP.conll.gz>
[4] <https://www.linguateca.pt/Floresta/ficheiros/amazonia.conll.gz>

- Modelo 4 - modelo em branco, acrescido do treino de 20 iterações dos "dados de treino".

Como pode-se observar, os primeiros resultados são bastante insatisfatórios, principalmente se compararmos aos que são obtidos para a língua inglesa. Assim, concluímos que seria necessário ampliar as bases de treinos, através de outros textos, realizamos os procedimentos iniciais e para nossa surpresa os resultados foram inesperados, cada um dos documentos teve como resultado da extração um texto diferente e na maioria inclusive, irreparável, visto que parágrafos tiveram a ordem das linhas invertidas, voltamos a testar ferramentas, mas todas passaram pelos mesmos problemas, foi ai que percebemos que cada um dos documentos foi gerado em uma versão diferente de PDF, que aparentemente faz com que as ferramentas não consigam seguir um padrão na extração. Devido a este acontecido optou-se por seguir com o treinamentos e extrair os textos manualmente, já que a tentativa de encontrar soluções para o ocorrido poderia consumir um tempo demasiado.

Observando a literatura, vimos que diversos trabalhos utilizam corpus para treinarem seus modelos, sendo assim optamos por utilizar o HAREM. Notamos que estrutura utilizada pelo HAREM difere da exigida pelo spaCy, sendo assim, buscamos possíveis trabalhos que já fizeram este mesmo processo, e conseguimos chegar no trabalho do Pires [2017], o mesmo desenvolveu alguns *scripts* para realizar o tratamento do corpus, porém quando ele realizou o trabalho a versão disponível do spaCy era a 1.7.2, sendo assim fez-se necessário a modificação dos mesmos para a versão a qual estávamos utilizando (spaCy 2.0.12).

Com o corpus do HAREM em condições para prosseguirmos, a próxima etapa foi definir como seriam os treinos, onde segundo Pires [2017], os melhores resultados para o spaCy e o HAREM, para a classificação de entidades nomeadas como notícias em uma base de dados chamada SIGARRA, foram obtidos com 110 iterações de treinamento, assim realizamos a criação de dois modelos, descritos abaixo, e os resultados podem ser visualizados na tabela 1

- Modelo 5 - novo modelo do HAREM, acrescido de 2 iterações com os "dados de treinamento";

- Modelo 6 - novo modelo do HAREM, acrescido de 110 iterações com os "dados de treino".

Tabela 1 – Resultados da Identificação - Primeira etapa

Modelo	Precisão	Cobertura	Medida-F
1	1,041	1,587	1,257
2	0	0	0
3	27,586	38,095	32,000
4	14,754	28,571	19,459
5	8,641	11,111	9,722
6	27,272	33,333	30,000

Fonte – Elaborado pelo autor

Devido a dificuldade de conseguir realizar uma extração automática de qualidade dos documentos e a alta demanda de tempo para fazer a marcação das entidades, optamos por trabalhar na extração dos dados que possuem uma padronização/formato definido sobre os documentos já processados, onde estes formatos são por exemplo datas, CNPJs e CPFs. Sendo esta operação inviável de ser realizada no spaCy por motivos como a forma de tokenização que o mesmo utiliza, pois está realiza a quebra da palavra quando encontra o caractere hífen (-) e o fato de tais entidades possuírem uma formatação padrão, que não justifica desperdiçar processamento e tempo de treinamento. Para realizar tal feito existe as estruturas Regex (expressões regulares), definido como um conjunto de regras/operações que visam extrair um conjunto de caracteres de uma entrada, logo foram gerados os Regex que seguem:

```
# CPF REGEX
"[0-9]{3}\.[0-9]{3}\.[0-9]{3}[\r\n]*\-[\r\n]*[0-9]{2}"
# CNPJ REGEX
"[0-9]{2}\.[0-9]{3}\.[0-9]{3}\/[0-9]{4}[\r\n]*\-[\r\n]*[0-9]{2}"
# DATA REGEX
"\D([0-9]{0,2}\/)?[0-9]{2}\/[0-9]{2,4}"
# MONEY REGEX
"((R\$\s)[0-9.]*\,?[0-9]*)"

```

Código–fonte 5.1 – Regex

Sabemos que estes primeiros testes são importantes para reconhecermos o nosso corpus e, também, para entendermos os pontos de melhoria dos modelos e dos treinos. Esses resultados estão muito aquém dos resultados encontrados na literatura, seja para a língua portuguesa, seja para a língua inglesa. Cabe ressaltar que o Modelo 3 utilizou os mesmos dados usados para treinamento e validação dos resultados, por isso ele chegou a um resultado próximo ao Modelo 6.

6 Considerações Finais

Compreendemos que a proposta de automatização de detecção de entidades nomeadas e seus vínculos em documentos escritos na norma culta da língua portuguesa é viável utilizando-se as ferramentas e bases de dados existentes. Entretanto, ainda não temos bons resultados com eles, visto que a identificação correta dessas entidades ainda não superou os 90% de acurácia, o que é um grande problema para a análise de documentos de caráter jurídico por parte das instituições envolvidas, como por exemplo documentos da Polícia Federal. Os modelos treinados, apesar de um melhor desempenho, também não são satisfatórios. Sabemos que é necessário identificar mais entidades em outros documentos para aumentar a qualidade do treinamento. Isto deve levar ao REN com maior precisão, em especial as entidades do tipo Pessoa.

Como trabalhos futuros, sugerimos a adaptação das outras bases de dados anotadas (FLORESTA, AMAZÔNIA e BOSQUE) para serem utilizadas como treinamento das principais ferramentas de NER: NERP-CRF, OpenNLP, spaCy e NLTK. Após os testes com todas as bases e ferramentas, buscaremos melhorias nos algoritmos para conseguirmos uma acurácia acima dos 95%, e uma cobertura mais alta, conforme já é apresentado na literatura para o idioma inglês.

Referências Bibliográficas

ADANIYA, M. H. A.; JR, M. L. P. Extração de informações na web. *Cadernos de Informática*, v. 4, n. 2, p. 27–34, 2009. Citado na página 18.

AMARAL, D. O. F. d. 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. Citado na página 27.

AMARAL, D. O. F. d. et al. Reconhecimento de entidades nomeadas na área da geologia: bacias sedimentares brasileiras. Pontifícia Universidade Católica do Rio Grande do Sul, 2017. Citado na página 18.

AMARAL, D. O. F. do; VIEIRA, R. Nerp-crf: uma ferramenta para o reconhecimento de entidades nomeadas por meio de conditional random fields. *Linguamática*, v. 6, n. 1, p. 41–49, 2014. Citado na página 26.

ATDAĞ, S.; LABATUT, V. A comparison of named entity recognition tools applied to biographical texts. In: IEEE. *Systems and Computer Science (ICSCS), 2013 2nd International Conference on*. [S.l.], 2013. p. 228–233. Citado na página 16.

CARVALHO, W. S. *Reconhecimento de entidades mencionadas em português utilizando aprendizado de máquina*. Tese (Doutorado) — Universidade de São Paulo, 2012. Citado 6 vezes nas páginas 19, 20, 21, 24, 26 e 27.

CHOI, J. D.; TETREAULT, J.; STENT, A. It depends: Dependency parser comparison using a web-based evaluation tool. In: *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. [S.l.: s.n.], 2015. v. 1, p. 387–396. Citado na página 26.

CONTIER, A.; PADOVANI, D.; NETO, J. J. Linguístico: Uma proposta de reconhecedor gramatical usando tecnologia adaptativa. *Revista de Sistemas e Computação, Salvador*, v. 2, n. 1, p. 70–81, 2012. Citado na página 26.

DOMO. *Data Never Sleeps 6 | Domo*. 2018. Disponível em: <https://www.domo.com/learn/data-never-sleeps-6>. Acesso em: 18 de agosto de 2018. Citado na página 13.

FONSECA, E. B.; CHIELE, G. C.; VANIN, A. A. Reconhecimento de entidades nomeadas para o portugues usando o opennlp. *Anais do Encontro Nacional de Inteligência Artificial e Computacional (ENIAC 2015), s. pp*, 2015. Citado 3 vezes nas páginas 15, 26 e 27.

FRED, L. *Reconhecimento de Entidades Nomeadas (NER) - O que é? Quais são as aplicações?* luisfredgs, 2018. Disponível em: <https://medium.com/luisfredgs/reconhecimento-de-entidades-nomeadas-ner-o-que-é-quais-s~ao-as-aplicaç~oes-cd1ab9a8a5e>. Acesso em: 18 de agosto de 2018. Citado 3 vezes nas páginas 13, 15 e 19.

MORAIS, N. A. Clustering de relacionamentos entre entidades nomeadas em textos com base no contexto. 2016. Citado na página 13.

OBSERVATóRIO de dados/Precisão e revogação. 2018. Disponível em: <https://pt.wikiversity.org/wiki/Observatório_de_dados/Precis~ao_e_revogaç~ao>. Acesso em: 22 de novembro de 2018. Citado na página 23.

PEREIRA, M. J. S. Processamento de linguagem natural para produtos de seguros. 2014. Citado na página 21.

PEREIRA, S. do L. Processamento de linguagem natural. 2016. Citado na página 17.

PIRES, A. R. O. Named entity extraction from portuguese web text. 2017. Citado 3 vezes nas páginas 13, 27 e 30.

ANEXO A – Códigos

```
import re
import sys
from subprocess import run, PIPE

lines = []
QTD_LINES_FILE = 0
QTD_LINES_READ = -1

def n_line(file):
  global QTD_LINES_READ
  QTD_LINES_READ += 1
  return re.sub(r"\s+", " ", re.sub(r"[U+25CFU+25AAU+2022]", " ", file.
    readline()))

def formating_text(infile, outfile):
  global QTD_LINES_FILE, QTD_LINES_READ
  f_in = "{f_in}".format(f_in=infile)
  f_out = "{f_out}".format(f_out=outfile)
  QTD_LINES_FILE = run("wc -l {f_file}".format(f_file=f_in),
              shell=True, stdout=PIPE)
  QTD_LINES_FILE = int(QTD_LINES_FILE.stdout.decode("utf-8").split(" ")[0])

  # CPF REGEX
  CPF_PATTERN = re.compile(r"[0-9]{3}\.[0-9]{3}\.[0-9]{3}\-[0-9]{2}")
  # CNPJ REGEX
  CNPJ_PATTERN = re.compile(
      r"[0-9]{2}\.[0-9]{3}\.[0-9]{3}\/[0-9]{4}\-[0-9]{2}")

  with open(f_in, "r") as file:
    with open(f_out, "w") as f:
      while QTD_LINES_READ <= QTD_LINES_FILE:
        line = ""
        line_1 = n_line(file)
        line_2 = n_line(file)

        while line_1.isspace():
          line_1 = line_2
          line_2 = n_line(file)

        line += line_1

        while QTD_LINES_READ <= QTD_LINES_FILE:
          if len(line) > 0:
```

```
43              check = [":", "Relacionados", "CPF/CNPJ"]
44              if any([any(x in line.rsplit()[-1] for x in check),
45                   all([any([CPF_PATTERN.match(line_1), CNPJ_PATTERN.match(
    line_1)]),
46                   any([CPF_PATTERN.match(line_2), CNPJ_PATTERN.match(line_2)
    ])])]):
47                lines.append(line.strip())
48                lines.append("\n")
49                if not line_2.isspace():
50                  lines.append(line_2.strip())
51                  lines.append("\n")
52                line = ""
53                line_1 = n_line(file)
54
55                while line_1.isspace():
56                  line_1 = n_line(file)
57                line_2 = n_line(file)
58                if not line_2.isspace():
59                  while not line_2.isspace():
60                    lines.append(line_1.strip())
61                    lines.append("\n")
62                    line_1 = line_2
63                    line_2 = n_line(file)
64                  lines.append(line_1.strip())
65                  lines.append("\n")
66                else:
67                  line += line_1
68                  lines.append(line.strip())
69                  lines.append("\n")
70                  line = ""
71            if line_2.isspace():
72              break
73
74            line += line_2
75            line_1 = line_2
76            line_2 = n_line(file)
77
78          if line != "":
79            lines.append(line.strip())
80            lines.append("\n")
81
82        f.writelines(lines)
83
84  # __main__
85  def main(args=None):
86    import argparse
87    P = argparse.ArgumentParser(description=__doc__)
```

```
88    P.add_argument("infile", type=str, default=None, help="Input file to
        process.")
89    P.add_argument("outfile", type=str, default=None, help="Output file.")
90    A = P.parse_args(args=args)
91
92    formating_text(**vars(A))
93    return 0
94
95 if __name__ == "__main__": sys.exit(main())
```

Código–fonte A.1 – heuristica.py