UNIVERSIDADE DE SÃO PAULO
FACULDADE DE FILOSOFIA, CIÊNCIAS E LETRAS DE RIBEIRÃO PRETO
DEPARTAMENTO DE COMPUTAÇÃO E MATEMÁTICA

ALEKSANDER TOMAZ DE SOUZA

Text chunking*: um método de *shallow parsing* para identificação de sintagmas nominais lexicais de textos em português do Brasil segundo o formalismo *Universal Dependencies

Ribeirão Preto–SP

2023

ALEKSANDER TOMAZ DE SOUZA

Text chunking*: um método de *shallow parsing* para identificação de sintagmas nominais lexicais de textos em português do Brasil segundo o formalismo *Universal Dependencies

Versão Revisada

Dissertação apresentada à Faculdade de Filosofia, Ciências e Letras de Ribeirão Preto (FFCLRP) da Universidade de São Paulo (USP), como parte das exigências para a obtenção do título de Mestre em Ciências.

Área de Concentração: Computação Aplicada.

Orientador: Prof. Dr. Evandro Eduardo Seron Ruiz

Ribeirão Preto–SP

2023

ALEKSANDER TOMAZ DE SOUZA

Text chunking: um método de *shallow parsing* para identificação de sintagmas nominais lexicais de textos em português do Brasil segundo o formalismo *Universal Dependencies*. Ribeirão Preto–SP, 2023.

95p. : il.; 30 cm.

Dissertação apresentada à Faculdade de Filosofia, Ciências e Letras
de Ribeirão Preto da USP, como parte das exigências para
a obtenção do título de Mestre em Ciências,
Área: Computação Aplicada.

Orientador: Prof. Dr. Evandro Eduardo Seron Ruiz

1. Análise sintática parcial. 2. Sintagmas nominais lexicais. 3. Dependência Universal

ALEKSANDER TOMAZ DE SOUZA

Text chunking: um método de *shallow parsing* para identificação de sintagmas nominais lexicais de textos em português do Brasil segundo o formalismo *Universal Dependencies*

Modelo canônico de trabalho monográfico acadêmico em conformidade com as normas ABNT.

Trabalho aprovado. Ribeirão Preto–SP, 02 de junho de 2023.

Orientador
Dr. Evandro Eduardo Seron Ruiz

Membro da banca
Dr. Ivan Rizzo Guilherme

Membro da banca
Dra. Vládia Célia Monteiro Pinheiro

Ribeirão Preto–SP
2023

"Àquele que é capaz de fazer infinitamente mais do que tudo o que pedimos ou pensamos"
(Bíblia Sagrada, Efésios 3:21)

Agradecimentos

Agradeço ao meu orientador e companheiro de pesquisa, o professor Dr. Evandro Eduardo Seron Ruiz por toda atenção e suporte, aos caríssimos professores Dr. Zhao Liang, Dr. Alexandre Martinez, Dr. Thiago Pardo, Dra. Roseli Romero, Dr. Eugênio Bucci e Dra. Vivian Batista da Silva. Aos monitores Guilherme Nardari, Roney Lira e Guilherme Martiniano de Oliveira, certamente grandes pesquisadores da computação. Ao pessoal dos grupos de Processamento de Linguagem Natural e Aprendizado Máquina do ICMC: Lucas, Thiago Vespa, Heber Gustavo, João Marcelo, Vitor, Emanuel, Xiomara e Pryscilla. Ao pessoal de Redes Complexas e Sistemas Computacionais Complexos II: aos caríssimos Michel, Welton, João Longo, Tatiana Pestana, Ana Caroline, José Andery, Luan Martins e Diana Arroyo. À Lúcia Akemi pela amizade e incansável dedicação ao Departamento de Computação e Matemática USP-RP, aos carríssimos Jalmei Andre Tomio e Rosangela Maria Laporti Seredynskyj também desse departamento. À professora Dra. Jaqueline Brigladore Pugliesi, minha primeira orientadora, uma inspiração no ensino. À solícita professora Dra. Magali Sanches Duran, pelos esclarecimentos. Aos solícitos professor Dr. Ruy Luiz Milidiu – PUC-Rio, professora Dra. Maria Cláudia Freitas – PUC-Rio e Dra. Cláudia Oliveira, pelas referências. Ao professor Dr. Ricardo Vencio, pela iniciativa em dispor seu trabalho a comunidade em *Machine Learning* pelo canal LabPIB. Ao professor Dr. Rodrigo Mello e as cativantes apresentações no canal *Machine Learning for You* (ML4U). Ao Departamento de Matemática, pelos convites de seminários estendidos à Computação. Ao professor Dr. Luis Arthur Pagani - UFPR, pelas interpretações acerca da gramática de dependência. A USP–RP pela resiliência durante o período de isolamento social em manter um ensino de excelência e pela resposta às demandas da nossa sociedade, debatendo e pesquisando acerca do agravo causado pela pandemia (COVID-19) nas suas diversas frentes. Meus agradecimentos também se estendem à FAPESP, USP e ao C4AI. Aos meus filhos Lucas e Pedro e a minha esposa Vania, meu irmão, Alyson e a Ariadina.

O termo análise pode ser compreendido como exame, pesquisa, verificação. Numa perspectiva científica, assume conteúdos semânticos peculiares em expressões como: 'análise funcional', 'análise harmônica', 'análise numérica' na matemática; 'análise elementar', 'análise conformacional' na linguagem química; 'análise econômica', 'análise estatística' para as ciências humanas e assim por diante (FERRARI, 1982).

Resumo

SOUZA, A. T. ***Text chunking*: um método de *shallow parsing* para identificação de sintagmas nominais lexicais de textos em português do Brasil segundo o formalismo *Universal Dependencies***. 2023. 91 p. Dissertação (Mestrado em Computação Aplicada) – Departamento de Computação e Matemática – Faculdade de Filosofia, Ciências e Letras de Ribeirão Preto, Universidade de São Paulo, Ribeirão Preto, 2023.

A análise sintática superficial, também conhecida pelo termo inglês 'shallow parsing', é um método computacional que identifica partes constituintes de uma frase (e.g.: verbos, substantivos e adjetivos) e as relaciona com estruturas gramaticais hierarquicamente superiores, os sintagmas (e.g.: nominais, verbais, preposicionais, entre outros). Este projeto aborda a identificação de um tipo específico de sintagma nominal definido como sintagma nominal lexical (SN_L), em textos escritos em português do Brasil, e anotados segundo o formalismo *Universal Dependencies* (UD). Os SN_L, devido a sua natureza discriminatória, assumem tipicamente funções temáticas ou semânticas e compõem um conjunto reservado de segmentos que chamamos de descritores textuais. Os SN_L são utilizados em várias tarefas de processamento de língua natural, tais como: extração e recuperação de informações, reconhecimento de entidades nomeadas, categorização de textos, análise de sentimentos, extração de fatos, extração de relacionamentos e sumarização de textos. Diferentemente da gramática de estruturas frasais, ou seja, a gramática de constituintes, a UD estabelece uma sintaxe de dependência entre palavras que pretende representar qualquer língua humana. A UD fundamenta-se na identificação, descrição, atribuição das relações de dependência existentes nos elementos de uma sentença, ou seja, seus termos e palavras. Neste projeto, recorremos a extração de SN_L sobre frases anotadas em UD de forma abstrata e inferencial utilizando algoritmos de Aprendizado de Máquina.

Palavras-chave: 1. Análise sintática parcial. 2. Sintagmas nominais lexicais. 3. *Universal Dependencies*.

Abstract

SOUZA, A. T. **Text chunking: a shallow parsing method for identification of lexical noun phrases of texts in Brazilian Portuguese according to the formalism Universal Dependencies**. 2023. 91 p. Dissertation (Master in Applied Computing) – Departamento de Computação e Matemática – Faculdade de Filosofia, Ciências e Letras de Ribeirão Preto, Universidade de São Paulo, Ribeirão Preto, 2023.

The superficial syntactic analysis, also known by the English term 'shallow parsing', is a computational method that identifies constituent parts of a sentence (e.g., verbs, nouns, and adjectives) and relates them with hierarchically superior grammatical structures, the phrases (nominal, verbal, prepositions, etc.). This project addresses the identification of a specific type of noun phrase defined as a lexical noun phrase (SN_L) in texts written in Brazilian Portuguese and annotated according to the Universal Dependencies (UD) formalism. The SN_L, due to their discriminatory nature, typically assume thematic or semantic functions and compose a reserved set of segments that we call textual descriptors. SN_L are used in various natural language processing tasks, such as information extraction and retrieval, named entity recognition, text categorization, sentiment analysis, fact extraction, relationship extraction, and summarization of texts. Unlike the grammar of sentence structures, that is, the grammar of constituents, the UD establishes a syntax of dependency between words that intends to represent any human language. The UD is based on the identification, description, and attribution of the dependency relationships existing in the elements of a sentence, that is, its terms and words. In this work, we extracted SN_L from sentences annotated in UD in an abstract and inferential way using Machine Learning algorithms.

Keywords: 1. Shallow parsing 2. Lexical noun phrase. 3.Universal Dependencies.

Lista de figuras

Lista de tabelas

Lista de abreviaturas e siglas

AM	Aprendizado Máquina
baseNP	*base Noun Phrase*
CoNLL	*Conference on Computational Natural Language Learning*
CoNLL-X	*Computational Natural Language Learning*
C4AI	*Center for Articial Intelligence*
DTC	*DecisionTreeClassifier*
EI	Extração de Informação
FN	Falso Negativo
FP	Falso Positivo
RFC	*RandomForestClassifier*
PoS	*Part-of-speech*
PCPT	*Perceptron*
PLN	Processamento de Língua (Linguagem) Natural
RFC	*RandomForestClassifier*
RI	Recuperação de Informação
SIGNLL	*Special Interest Group on Natural Language Learning*
SN_L	Sintagma Nominal Lexical
TBL	*Transformation-based Learning*
UD	*Universal Dependencies*
VN	Verdadeiro Negativo
VP	Verdadeiro Positivo
XGBC	*XGBClassifier*
XGBRFCl	*XGBRFClassifier*

Sumário

1

Introdução

Na área de pesquisa de Processamento de Línguas Naturais (PLN), a análise sintática, ou *parsing*, é definida como a atividade de atribuir uma estrutura sintática implícita a uma sentença. Por meio do estudo sintático de uma sequência de palavras pode-se determinar padrões e compreender significados dos termos contidos numa sentença. Assim, é afirmado que a sintaxe produz uma estrutura lógica compreensível e, por conseguinte, identificável de palavras na sentença (JURAFSKY; MARTIN, 2021).

Nesse mesmo referencial bibliográfico, os autores Jurafsky e Martin ainda apontam que essa estrutura sintática, quando analisada computacionalmente, pode ser utilizada em aplicações de PLN tais como: verificação gramatical, análise semântica e perguntas e respostas (*question and answering*).

Com isso, temos que a análise sintática de uma frase implica na interpretação de uma sequência linear de palavras para compor estruturas que mostram como essas palavras se relacionam entre si. A literatura especializada descreve basicamente dois tipos de análise sintática:

- A análise sintática completa, ou *full parsing*, que analisa a função de cada termo de uma oração e também as regras que regem a construção de frases nas línguas naturais; e

- A análise sintática parcial, também chamada de *shallow parsing* (SP), que, diferentemente da análise completa, consiste na tarefa de recuperar apenas uma quantidade limitada de informações sintáticas de frases de linguagem natural (HAMMERTON et al., 2002).

Destacamos que muitas tarefas em PLN prescindem de uma análise sintática completa, ou seja, muitas tarefas de processamento de linguagem simplesmente não requerem árvores de análise completas e complexas para todas as entradas, o que, consequentemente,

reduz a complexidade computacional da aplicação (CHURCH; PATIL, 1982). Jurafsky em seu livro (JURAFSKY; MARTIN, 2021), na página 270, exemplifica que:

> (...) os sistemas de extração de informações geralmente não extraem todas as informações possíveis de um texto; eles simplesmente identificam e classificam os segmentos em um texto que provavelmente contêm informações valiosas. Da mesma forma, os sistemas de recuperação de informação podem optar por indexar documentos com base em um subconjunto selecionado de constituintes encontrados em um texto. (tradução livre)

Nesse contexto, trazemos a proposição feita durante a *Conference on Computational Natural Language Learning* (CoNLL) 2000, que definiu o *text chunking* como 'a fragmentação de um texto em conjuntos de palavras sintaticamente correlatas', (SANG; BUCHHOLZ, 2000), o que declara, não somente, um método computacional de análise, mas, também, destaca a existência de sequências de palavras que reportam a descrição de excertos, ou trechos, até então compreendidos como estruturas sintagmáticas naturais da gramática de constituintes ou sintagmas.

A identificação e extração de descritores textuais, expressões nominais, termos índices, ou, ainda, indexadores textuais consiste numa das atividades propostas pelo PLN. Dados textuais compõem atualmente uma considerável parte da produção massiva de dados não-estruturados, e, nesse contexto, encontrar meios de selecionar documentos textuais de interesse impõe, cada vez mais, a implementação de critérios que atendam a uma série de passos cientificamente fundamentados frente as diferentes teorias.

A pesquisa de Ophélie Lacroix (LACROIX, 2018) busca a identificação de marcações sintagmáticas nominais sobre textos anotados pelo formalismo gramatical de dependência universal, *Universal Dependency* (UD), proposta originalmente por Nivre *et al.* (NIVRE; MCDONALD, 2008). Assim, por meio das marcações de natureza sintática desse recente formalismo, enseja a recuperação de sintagmas frasais naturais da gramática de constituintes. Cabe aqui mencionar que as duas teorias consideradas nessa pesquisa, a gramática de constituinte e a gramática de dependência, retratam as mesmas estruturas sintáticas sob perspectivas diferentes (RAMBOW, 2010).

Dessa forma, visamos identificar segmentos sintagmáticos denominados sintagmas nominais lexicais (doravante SN_L) propostos por Oliveira e Freitas (OLIVEIRA; FREITAS, 2006). Estes SN_L são caracterizados por possuírem uma configuração impermanente que assume em suas formas mais extensas algumas propriedades dos chamados sintagmas nominais complexos (SNC).

Os Sintagmas Nominais Lexicais (SN_L) são apresentados nesta pesquisa como descritores textuais, ou seja, essas estruturas sintagmáticas são apontadas como elementos

de síntese textual. Podemos entendê-los como um sintagma nominal (SN) mais complexo por permitir em sua composição sintagmas adjetivais e preposicionais e, dentre outras características, essa formação destaca-se por exigir um núcleo estritamente substantivo.

Esses segmentos textuais apontados nos remetem a identificadores e/ou especificadores, para as mais diversas aplicações, sendo eles elementos por meio dos quais uma série de atividades relevantes podem ser tratadas tanto no Processamento de Linguagem Natural quanto em outros campos científicos relacionados a Extração de Informação (EI) ou Recuperação de Informação (RI). Por exemplo: i) **O lenço** caiu. ii) **O lenço perfumado** caiu e iii) **O lenço perfumado de Cecília** caiu. Permitindo num mesmo cenário distinguir diferentes atores que possam ter um constituinte comum em sua caracterização.

O estabelecimento de análise sintática segundo o formalismo UD, por exemplo, introduz uma perspectiva de estruturas regidas pelas relações de dependências existentes em uma sentença escrita. Nessa gramática, a ausência de marcações sintagmáticas persuade a uma pesquisa que permita afirmar ou não a existência de correlação entre as relações de dependência universal e sua morfossintaxe própria com a composição sintagmática. Dessa maneira, uma questão de pesquisa que se nos apresenta é: as marcações UD possuem correlação frente as estruturas sintáticas como os sintagmas nominais lexicais? Uma vez recuperadas, por meio dos atributos UD, as classificações sintagmáticas podem trazer a representação dessas estruturas sintáticas?

Para aplicação do método de *shallow parsing* recorremos algoritmos de aprendizado de máquina (AM), sendo um tradicional e outros modelos, ou *emsembles*, tidos como mais avançados. Buscamos, consequentemente, reportar seus desempenhos frente a apresentação de dados extraídos do formalismo UD.

1.1 Objetivo geral

Diante do exposto acima, pretendemos identificar um tipo específico de sintagma nominal chamado sintagma nominal lexical (SN_L) em textos anotados no formalismo de dependência universal (NIVRE; MCDONALD, 2008). Nesse mesmo sentido, nosso objetivo principal de pesquisa é o reconhecimento de SN_L em frases escritas em português e anotadas segundo o formalismo UD para auxiliar no processo de recuperação de informação e reconhecimento de termos primários os quais, eventualmente, poderão ser usados, por exemplo, na indexação e na pesquisa por documentos.

1.2 Objetivos específicos

Como objetivos secundários relacionamos:

- Modelar um corpus que possua marcações dos SN_L e *tags UD*;
- Estabelecer um *baseline* de resultados de extração automática de SN_L sobre a porção de avaliação de textos do *corpus* Bosque UD usando o método TBL, *Tranformation Based Learning*;
- Avaliar o desempenho do TBL quanto a recuperação de SN_L usando rótulos morfossintáticos (UD PoS Tag) e marcações de relações de dependência *Universal Dependency*;
- Identificar os principais padrões de anotação morfossintática e de dependência universal que marcam os sintagmas nominais;
- Aplicar modelos de AM baseados em outras aproximações;
- Estabelecer um comparativo entre as medidas a partir dos resultados obtidos pelos diferentes métodos de AM.

1.3 Organização

No próximo capítulo, Capítulo 2, apontamos os principais referenciais teóricos que embasam esse projeto de pesquisa. Em seguida, no Capítulo 3 destacamos alguns trabalhos que compreenderam o método de *shallow parsing*. Após isso, no Capítulo 4 descrevemos nossos passos de pesquisa quanto aos recursos, métodos e experimentos realizados. Mais adiante, no Capítulo 5 desta dissertação, apresentamos os resultados e considerações alcançados. Por fim, Capítulo 6 deixamos nossas impressões acerca do trabalho, bem como as contribuições que podem decorrer dele. Os Anexo A e Anexo B trazem uma estatística descritiva dos dados do *corpus* e nos Anexos C e D nossos resultados de métricas preliminares.

2

Referencial teórico

Neste capítulo abordamos os principais referenciais teóricos sobre análise sintática parcial, sintagmas nominais e sintagmas nominais lexicais, incluindo os trabalhos que formam o referencial teórico da nossa proposta.

2.1 *Shallow parsing*

Hammerton (HAMMERTON et al., 2002) esclarece que a análise sintática parcial deve ser considerada como referência a um conjunto de métodos que procura recuperar algumas informações mediante a ausência de outras. Esses pesquisadores indicam que essa aproximação permite auxiliar atividades em PLN, tais como: *part-of-speak tagging* (PoS) – inferir a classe morfossintática de uma palavra; *text chunking* – identificação de sintagmas nominais, verbais, adjetivais, dentre outros; e estabelecimento de relações – discriminar as funções que os sintagmas assumem diante o verbo num determinado contexto: sujeito, objeto, entre outros.

Com esse entendimento, ao alinhar o artigo de Hammerton (HAMMERTON et al., 2002) e recuperar as definições da obra de Jurafsky e Martin (JURAFSKY; MARTIN, 2021), onde se destaca que *text chunking*:

> É o processo de identificar e classificar os segmentos planos e não-sobrepostos de uma sentença que constituem sintagmas não-recursivos básicos, correspondentes à principal classe de palavras de conteúdo neles inserida: sintagma nominal (SN), sintagma verbal (SV), sintagma adjetival (SADJ) e sintagma preposicional (SPP) (JURAFSKY; MARTIN, 2021).

Com a devida vênia, podemos tomar que mesmo que a hierarquia da gramática de constituintes expressa que os sintagmas frasais (SN, SV, DADJ, SPP) sejam compostos

por constituintes sobrepostos, o que permite, por exemplo, uma análise recursiva nesses segmentos, este não é um fato, no sentido estrito, aplicável em nossa pesquisa.

Encontra-se na literatura, outra forma de se referir a esse modo de seleção de sintagmas frasais. Ophlélie Lacroix (LACROIX, 2018) diz que *syntactic chunking* ou *chunking*:

> ... consiste na identificação de grupos de palavras (consecutivas) em uma sentença que constituem sintagmas (*phrases*) (por exemplo, *noun-phrases, verb-phrases*). *Chunking* pode ser visto como uma tarefa de análise superficial entre a marcação *PoS tagging* e a análise sintática (N.T.: completa). *Chunking* é conhecido como uma etapa de pré-processamento relevante para a análise sintática.

Na definição acima, Lacroix resgata a definição inicialmente proposta de Abney (ABNEY, 1992) quanto ao *shallow parsing* quando diz que ela corresponde 'a tarefa de dividir um texto em segmentos não-sobrepostos sintaticamente correlatos' (tradução livre).

Isto posto, compreendemos haver concordância entre o que afirma Lacroaix e as definições elaboradas por Hammerton *et al* (HAMMERTON et al., 2002), Jurafsky e Martin (JURAFSKY; MARTIN, 2021), como também Abney (ABNEY, 1992), acerca da atividade de selecionar estruturas frasais de textos escritos em língua natural, ou seja, *text chunking*, conceituando um método específico, o *shallow parsing*.

Dos sintagmas resultantes do *text chunking*, podem-se extrair conteúdos relevantes para análise textual; perpassando ocorrências nas quais a entrada de dados seja de baixa qualidade, como, por exemplo, em 'frases que podem ter palavras repetidas, palavras faltantes ou quaisquer outros erros lexicais e sintáticos' (LI; ROTH, 2001).

Assim sendo, aliando essas considerações a definição de *shallow parsing* vista no Capítulo 1, página 27, a princípio, nos permite destacar algumas características da metodologia necessária para executar a tarefa de análise sintática parcial, segundo as perspectivas computacional e linguística, as quais naturalmente enumeramos a seguir:

1. **Eficiência**, uma vez que deve requer menor processamento computacional do que a análise sintática completa, ou seja, é esperado que, comparativamente, a análise parcial exija menos recursos computacionais;

2. **Rapidez**, espera-se uma análise rápida por atuar em informações mais superficiais, o que pode ser útil em situações em que o tempo é um fator crítico;

3. **Redução de ruído**, ou seja, este tipo de análise deve eliminar informações desnecessárias e focar nas informações mais relevantes para a tarefa;

4. **Identificação de padrões**, espera-se que este tipo de análise reconheça termos, palavras ou estruturas sintáticas comuns, o que pode também auxiliar em tarefas como a classificação de texto;

5. **Melhoria do desempenho de outras tarefas de PLN**, a análise sintática parcial pode ser usada como uma etapa de pré-processamento para outras tarefas complexas de PLN, tais como a análise semântica, e;

6. **Redução de ambiguidade** dada a possibilidade de identificação de relações entre palavras e estruturas sintáticas, a análise sintática parcial pode ajudar a melhorar a compreensão do texto e a precisão das análises subsequentes.

Logicamente, umas ou outras dessas características podem se destacar mais a depender da aplicação em que é empregada e os recursos envolvidos.

2.2 Sintagmas nominais

Discorrendo sobre os sintagmas, vemos que esses segmentos textuais são considerados por Kroch e Silva (SILVA; KOCH, 2012) como:

> '(...) conjuntos de elementos que constituem uma unidade significativa na oração, mantendo relações de dependência e de ordem entre si. Organizam-se em torno de um elemento fundamental, denominado núcleo, que pode, por si só, construir um sintagma.'

Dentre os sintagmas existentes, os nominais são aqueles compostos por termos de teor substantivo, sendo esse, o substantivo, o elemento fundamental de sua estrutura. Tradicionalmente, nomes e pronomes também assumem esse papel.

Sintagmas nominais (SN) têm sua importância analítica por discriminar elementos com sentido substantivo que assumem funções temáticas ou funções semânticas, respectivamente sujeito–objeto, ou agente–instrumento. Segundo Ramsden (RAMSDEN, 1974), termos que refletem características de um vocabulário regido num domínio sintático fazem parte do que podemos compreender como uma linguagem de indexação, ou seja, atuam com indexadores textuais representativos (KURAMOTO, 1996). Assim, essas expressões lexicais atendem como elemento de síntese.

Algumas ferramentas conhecidas como analisadores, com diferentes aplicações, têm nas sequências de teor substantivo, SN, elementos que permitem estabelecer relações entre dados textuais distintos. São elencadas no trabalho de Maia e Souza (MAIA; SOUZA,

2010) como importantes indexadores textuais para o português e a estes acrescemos o OGMA, o analisador dos próprios autores. Veja a Tabela 1, a seguir.

Tabela 1 – Relação de ferramentas de indexação textual para o português

Ferramentas	**Referências**
VISL	Bick (1996)
PoSiTagger	Aires (2000)
Curupira	Martins (2002)
Grammar Play	Othero (2004)
OGMA	Maia (2010)
LX-Tagger / LX-Suite	Branco (2014)

Se considerarmos os SN como descritores texturais, podemos citar alguns cenários e aplicações que derivam do *text chunking* e se estendem à área de extração e recuperação de informações.

A EI (extração de informação) consiste na obtenção de conteúdos relevantes de um ou mais documentos (COWIE; LEHNERT, 1996). Kusmerik define EI como: 'a tarefa de identificar fragmentos específicos de um documento que constituem o núcleo de seu conteúdo semântico' (KUSHMERIK, 1999). Nesta atividade estão inseridas tarefas como: extração de entidades ou reconhecimento de entidades nomeadas (WU; ZHAO; XU, 2003), categorização de textos (ZHAI; MASSUNG, 2016), agrupamentos (PHRIDVIRAJ; RAO, 2015), extração de fatos, extração de relacionamentos (KRUG; MERGEN, 2013) e sumarização de texto. Todas essas tem como característica principal apreender o 'que é essencial', por meio da faculdade de 'destacar o que é relevante' ao leitor (RINO; PARDO, 2003).

A RI (recuperação de informação) é uma área da Ciência da Computação que têm por objetivo selecionar um conjunto relevante de documentos para o usuário de acordo com a requisição apresentada por este (MOOERS, 1951). Essa tarefa pode ser relacionada ao reconhecimento de termos primários, busca e indexação de documentos. Assim, assumimos que os SN, pela sua natureza discriminatória, são elementos essenciais para composição de funções e expressões de busca que recuperam estes indexadores.

Ramshaw e Marcus (RAMSHAW; MARCUS, 2002), em seus estudos, descreveram o *text chunking* como a fragmentação de um texto de modo que haja a divisão da sentença em segmentos não-sobrepostos com base em uma análise bastante superficial. Com tal preceito, estabeleceram o segmento textual chamado **baseNP** (*base noun phrase*) que essencialmente representa trechos não-recursivos de *noun phrases*, ou seja, os sintagmas nominais. Essas *phrases* são caracterizadas por terem o substantivo como seu termo principal, podendo

assumir a função de sujeito ou objeto com relação ao verbo da oração (NEVES, 2000).

2.3 Sintagmas nominais lexicais

Inspirados nos trabalhos pregressos citados acima sobre o conceito de sintagma nominal (SN), Oliveira e Freitas (OLIVEIRA; FREITAS, 2006) apontam para um tipo específico de SN que fora aplicado à recuperação de informação. Trazemos para essa pesquisa este tipo específico de sintagma nominal, definido como sintagma nominal lexical. Esse tipo de sintagma é constituído por um núcleo estritamente substantivo, seguido de seus especificadores, modificadores e complementos (SVOBODOVÁ, 2014).

Os SN_L são discernidos dentre os sintagmas nominais por tomarem apenas substantivos como elemento principal e por permitirem, em sua extensão, sintagmas adjetivais e preposicionais. Os fundamentos teóricos para a criação dos SN_L foram aplicados em razão dos estudos sintáticos como os contidos no livro *Syntactic Theory and the Structure of English*, de Andrew Radford (RADFORD, 1981), em que a análise da estrutura sintática destaca apenas léxicos substantivos como núcleo, o que é exemplificado na Figura 1.

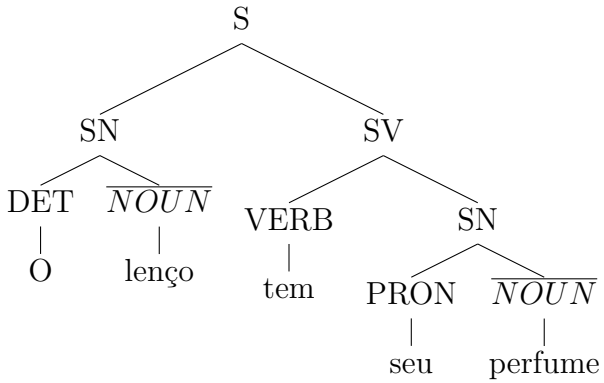

Figura 1 – Árvore de estrutura sintática com SN_L em destaque

Nesta Figura 1, os SN_L são representados por meio da marcação de uma barra sobre a etiqueta morfossintática dos termos nucleares dos respectivos segmentos sintagmáticos nominais e, nesta demonstração, os aproxima dos SN. Essa escolha, segundo as autoras, implica em um maior potencial de identificação de conteúdos discriminatórios para tarefas recuperação de unidades de informação textual (OLIVEIRA; FREITAS, 2006).

O núcleo do sintagma nominal lexical (SN_L) corresponde ao elemento fundamental e único, de significativo teor substantivo. Esse núcleo determina a concordância interna da expressão na qual está inserido. Para efeitos de definição, excluem-se, nesse contexto, construções elípticas[1].

[1] Na frase 'É melhor um pássaro na mão do que dois [pássaros] voando', a elipse ocorre pela supressão da palavra *pássaros*, no caso, um substantivo.

O núcleo do sintagma nominal lexical pode vir associado a outros elementos de interesse nessa pesquisa tais como se vê em negrito: especificadores ou determinantes (**O** nome da rosa), modificadores (Meu ipê **amarelo**) e complementos (A queda **da Bastilha**). Os modificadores e complementos do SN_L são, respectivamente, constituídos por sintagmas adjetivais e preposicionais. Orações relativas e apostos aqui são descartados. Os especificadores compreendem artigos, pronomes demonstrativos, possessivos e indefinidos; ou quantificadores, como os numerais.

Uma forma de destacarmos o objetivo do projeto, pode ser demonstrado no texto: 'Ouviram do Ipiranga **as margens plácidas**'. Neste texto, o termo '**margens**', em destaque, é o núcleo do sintagma nominal lexical, o termo '**as**' um especificador e o termo '**plácidas**' deve ser entendido como modificador do termo principal.

2.3.1 Exemplos de sintagmas nominais lexicais

A identificação de SN_L, como no exemplo 7 abaixo, pode ser trivial. Nestes exemplos SN_L estão marcados em negrito. Percebe-se, no entanto, que esta identificação pode se tornar uma atividade complexa em ocorrências, tais como:

- Segmentos nominais coordenados que devem ser compreendidos como SN_L independentes, como no exemplo 8, ou em
- Frases de maior extensão para as quais determinar os limites do SN_L envolve compreender segmentos sintagmáticos adjetivais e preposicionais (modificadores e complementos). Veja o exemplo 9.

7. **A caneta** é esferográfica.
8. **Caneta** e **papel** para escrever.
9. **Caneta esferográfica *Montblanc*** para escrever em **papel apergaminhado de cor sépia**.

Questões que envolvem entidades nomeadas, tais como, nomes próprios, nomes de entidades governamentais ou instituições e localizações geográficas, devem ser consideradas um único elemento. Veja o exemplo 10, em que '**João Pessoa**' deve ser entendido como unidade, ou seja, um núcleo de SN_L:

10. **João Pessoa** é capital de um estado brasileiro.

A restrição de núcleo lexical desconsidera como SN_L segmentos em que pronomes substantivos (exemplo 11) e numerais exerçam função de elemento principal (veja

exemplo 12) por serem 'referência anafórica a outro elemento lexical ou oracional no discurso' (OLIVEIRA; FREITAS, 2006).

11. **Ela** é capital de um estado brasileiro.

12. **Os três** são bons livros.

Quanto a segmentos em que numerais exerçam funções de modificadores ou complementos, eles serão composição junto ao termo principal a que se referem para formação do SN_L. Veja o exemplo 13.

13. **Os Três Mosqueteiros** é um romance histórico escrito por Alexandre Dumas.

Outro caso a ser considerado relaciona-se a segmentos textuais nos quais a coordenação de núcleos pode gerar ambiguidade. Não se permite estabelecer se os '**argumentos**' são também '**incontestáveis**', ou se somente o termo '**fatos**' possui esse modificador. Veja o exemplo 14, a seguir:

14. Os **argumentos** e **fatos** incontestáveis.

Além desses pontos, estamos cientes que a língua possui propriedades que tornam a tarefa de identificação de um descritor textual difícil. Esperamos compreender e em parte perpassar alguns desses problemas com a uso de dependência universal para redução de taxa de ruído e silêncio, como as encontradas por Kuramoto (KURAMOTO, 1996). Notem a polissemia, nos exemplos 15, 16, 17; a sinonímia, nos exemplos 18 e 19, bem como a disposição das palavras, nos exemplos 20 e 21.

15. **Letra**: com significado de elemento do alfabeto.

16. **Letra**: com significado de texto de uma canção.

17. **Letra**: com significado de título de crédito.

18. **Caneta**: aquilo que usa para escrever.

19. **Pena**: uma referência semântica a caneta.

20. Vítimas de crimes **juvenis**. (KURAMOTO, 1996)

21. Vítimas **juvenis** de crimes. (KURAMOTO, 1996)

2.3.2 Revisão bibliográfica sobre os SNL

A revisão bibliográfica sistemática permite estabelecer protocolos para identificação e interpretação de pesquisas relevantes a um determinado alinhamento, considerando aspectos relacionados tanto ao objeto ou ao método quanto, também, aos critérios do contexto em que se é aplicado (KITCHENHAM, 2004).

Por meio desses apontamentos, frente ao objeto de pesquisa, propriamente os SN_L, algumas das questões que podem ocorrer são: que outro tipo de sintagma nominal de interesse aplicado existe e qual sua aproximação com o sintagma nomina lexical; quais outros estudos foram realizados envolvem os SN_L ou correlatos no português, ou em língua inglesa, dado o predomínio de produções acadêmicas nesse idioma? Seguimos com a revisão bibliográfica.

Elhadad em seu artigo (ELHADAD, 1996) que trata, dentre outros temas, do planejamento de construção de sentenças e sintagmas complexos, refere que as estruturas de SN é menos compreendida e consequentemente mais complexa do que das sentenças. Ele descreve que a sintaxe sentencial permite tipicamente uma estrutura de relações predicado-argumento, enquanto as relações sintáticas dos SN complexos exigem uma hierarquia, ou árvore sintática, em sua estrutura. Quanto a essa questão, os elementos desses SN são representadas por um conjunto discreto e, portanto, representam um conjunto de cardinalidade limitada.

A literatura linguística enumera uma série de estudos acerca dos SN. Quanto a essas aproximações, estes elementos levam nomenclaturas e apresentam interpretações diferentes que, num ou noutro momento, podem ter pontos comuns, apesar de seus contextos serem singulares. Encontramos uma demonstração deste aspecto nos sintagmas nominais complexos (SNC). Os SNC são caracterizados pela presença de diferentes modificadores sintáticos como adjetivos, substantivos, sintagmas preposicionais e orações relativas. Esses sintagmas diferem do SN_L por permitirem uma composição mais ampla de elementos, no caso a ocorrência de orações relativas, assim como a exigência de um núcleo composto, permitindo classificar, num único sintagma complexo, segmentos onde há coordenação de núcleos.

O estudo de aspectos sintáticos-semântico-discursivos, respectivamente internos, de sentido e de contexto ou externos, da estrutura dos sintagmas nominais complexos, quiça SN_L, fora abordado por Givón (GIVÓN, 2001). Neste trabalho, o autor pondera a complexidade dessas estruturas conforme a quantidade de sintagmas subordinados ao termo nuclear. Havendo diferentes arranjos que possam ser consideradas em sua definição, o autor relacionou uma série de critérios que permitem representar a densidade e diversidade dessas estruturas, criando, assim, uma metodologia para análise descritiva e interpretativa dos tais. Para ele:

> A complexidade sintática de um SN manifesta-se em diversos níveis da estrutura. Para começar, a mera presença de um modificador já revela a existência de um nível hierárquico extra, no qual o substantivo núcleo e o modificador são nós irmãos sob o nó SN mais alto. A complexidade, então, existe quando é adicionado um nó ao substantivo núcleo.(GIVÓN, 2001). (Tradução livre)

Da Silva (SILVA, 2020) destaca que as estruturas dos SN possuem um gradiente de complexidade na qual a distinção está mais explícita se tomarmos a divisão discurso-textual, ou fala-escrita. Pelos estudos de sequências de textos de Adam (ADAM, 2011) que compreende a classificação de segmentos textuais mais extensos que frases, como: narrativa, descritiva, argumentativa, de diálogo e explicativa, a autora aponta um tipo de sintagma nominal chamando sintagma nominal complexo. Neste artigo, ela identifica que este tipo de sintagma prevalece entre as estruturas nominais nos seguimentos classificados como explicativos ou séries explicativas, representados regularmente por resumos e títulos de artigos e trabalhos acadêmicos.

Beijsterveldt e Hell (BEIJSTERVELDT; HELL, 2010) trataram acerca de textos narrativos e explicativos e recorreram às observações de ocorrências e construções de sintagmas nominais complexos como ponderador cognitivo quanto a aspectos da compreensão e descrição em portadores de deficiência auditiva.

Godby (GODBY, 2002) traz outro contexto linguístico e filosófico acerca da SN que possuem composição complexa. A autora identifica propriedades que levam a uma distinção quanto aos fundamentos requeridos para compreensão dessas estruturas, uma remete a uma composição semântica dessas construções definindo o que chama de sintagmas nominais sintáticos e outras levam a relações entre seus léxicos nomeando os como sintagmas nominais lexicalizados:

> ... os sintagmas nominais sintáticos são interpretados por regras semânticas composicionais, enquanto os sintagmas nominais lexicalizados sempre têm significado idiossincrático, podemos traçar uma distinção que explica a diferença fundamental em seu uso. (tradução livre)

Dessa forma, podemos inferir que a composição dos sintagmas nominais lexicais remete a um complexo de maior teor descritivo do sintagma nominal, ou seja, é de instrumentalização contextual ou semântica, bem como sintática. Mais claramente, quando compomos a expressão 'conclusões lógicas' a classe de conceitos denotada pela palavra 'conclusões' cruza com a classe de conceitos considerados 'lógicos', e o significado de 'conclusões lógicas' reside nessa conjuntura, ou seja, há regras explícitas que permitem a compreensão do significado dessa expressão que podem se derivar dos constituintes que a formam. Já em segmentos como 'casa de verão', 'sistema solar' e 'erosão recorrente'

não há uma regra aquém da composição semântica que permita definir o ente que se fala, necessitando assim uma experiência anterior que nos remeta ao seu significado.

2.4 Critérios da análise sintática

A atuação daqueles algoritmos computacionais automatizados, tanto sob a gramática de constituintes em que as estruturas sintáticas são expressas pelos sintagmas, quanto para o formalismo de dependência universal no qual as relações se estabelecem de acordo regras predicador–argumento para compor a árvore sintática, deve-se seguir um critério para seleção dos segmentos textuais internos à sentença. Hjelmslev (HJELMSLEV, 1975) traz a seguinte indicação para os métodos que se propõe a definir e descrever segmentos textuais:

> (...) o essencial não é dividir um objeto em partes, mas sim adaptar a análise de modo que ela seja conforme às dependências mútuas que existem entre essas partes, permitindo-nos prestar contas dessas dependências de modo satisfatório. (...) tanto quanto suas partes, o objeto examinado só existe em virtude desses relacionamentos ou dessas dependências; a totalidade do objeto examinado é apenas a soma dessas dependências, e cada uma de suas partes define-se apenas pelos relacionamentos que existem: i) entre ela e outras partes coordenadas, ii) entre a totalidade e as partes do grau seguinte, iii) entre o conjunto dos relacionamentos e das dependências e essas partes

Acresça-se que o resultado de uma análise sintática parcial, numa ou noutra teoria, deve representar o mesmo segmento textual. Isso posto, notamos que a gramática de constituintes ao descrever as relações de domínio imediado e de precedência linear existentes entre os termos das sentenças, como argumenta Partee *et al.* (PARTEE; MEULEN; WALL, 1994), não descarta a indicação de Hjelmslev quanto aos critérios de fragmentação e análise, o que a harmoniza com o nosso entendimento de sintagma apresentado no capítulo introdutório, que remete a uma unidade elementar não sobreposta. Vejamos aqui alguns critérios linguísticos aplicados aos constituintes frasais (sintagmas) que podem nos levar a uma intuição final:

Mobilidade: afirma que uma sequência de palavras forma um constituinte e que esse pode ocupar diferentes lugares na frase. Podemos observar a característica de mobilidade consoante o que se observa no constituinte '**durante a aula**' nas demonstrações 22, 23, 24 e 25, abaixo.

22. O aluno escreveu seus argumentos no caderno **durante a aula**.

23. **Durante a aula**, o aluno escreveu seus argumentos no caderno.

24. O aluno, **durante a aula**, escreveu seus argumentos no caderno.

25. O aluno escreveu, **durante a aula**, seus argumentos no caderno.

Inseribilidade: estabelece a impossibilidade de inserção de um constituinte noutro constituinte, ou seja, não pode haver interrupção na sequência do constituinte, como representado no exemplo 26.

26. O aluno escreveu *seus* **no caderno** *argumentos* durante a aula.

Nota-se estranheza na construção acima na qual o constituinte *seus argumentos* é interrompido com o posicionamento do constituinte **no caderno** em sua estrutura.

Enunciabilidade: manifesta-se pela possibilidade de isolamento, quando se diz que, num dado contexto, o constituinte satisfaz a coerência textual sem que haja redação da frase completa. Esta característica é demonstrada no exemplo 27, ou no contexto escolar, como pode-se deduzir como se fosse redigida em 28. Reforçamos que a presença do constituinte **no caderno** pode ser considerada como implícita na frase 27.

27. Onde o aluno escreveu seus argumentos?

28. O aluno escreveu seus argumentos **no caderno**.

Ainda, segundo Grahl disserta (GRAHL, 2009), há outros critérios (**Coordenabilidade** e **Substituição Pronominal**) que podem servir como recursos para confirmação dos anteriores quanto a definição dos constituintes da sentença.

Por fim, quanto aos critérios para análise sintática, as questões relacionadas aos estudos de Greenberg (GREENBERG, 1963), reafirmados Velupillai (VELUPILLAI, 2012), reconhecem a disposição dos elementos sintáticos existentes em diversas línguas, o que remete a incerteza. As seis ocorrências de ordenação dos termos de uma sentença, exemplos 29 ao 34:

29. Sujeito–Verbo–Objeto (SVO) *A menina pega o livro.*

30. Sujeito–Objeto–Verbo (SOV) *A menina o livro pega.*

31. Verbo–Sujeito–Objeto (VSO) *Pega a menina o livro.*

32. Verbo–Objeto–Sujeito (VOS) *Pega o livro a menina.*

33. Objeto–Sujeito–Verbo (OSV) *O livro a menina pega.*

34. Objeto–Verbo–Sujeito (OVS) *O livro pega a menina.*

À essas, acrescemos a existência de línguas onde não há predominâncias de alguma das disposições anteriormente relacionadas, sendo denominadas línguas de tipologia: Nenhuma Ordem Dominante (NDO) (DRYER; HASPELMATH, 2013). Essas formas distintas de ordenamento sintático refletem o modo no qual as estruturas do pensamento, ou modelos de língua, são organizados para se expressarem nos idiomas estudados.

2.5 Sintaxe de constituintes

A análise sintática atualmente pode estabelecer suas estruturas de acordo com dois fundamentos distintos: a sintaxe de constituintes que recorre a chamada estrutura frasal (*phrase structure*), e sobre o fundamento da dependência universal que veremos adiante, na página 43. Foquemos na sintaxe de constituintes.

Assim como analise sintática completa, a análise sintática parcial, em PLN, segue, tradicionalmente, a gramática de constituintes. Vemos algumas razões para essa abordagem: i) a tradição dessa gramática no campo linguístico, também dita, gramática de estruturas frasais; ii) suas unidades elementares possuem propriedades recursivas, o que permite implementações computacionais por meio de vários paradigmas, pois, como exemplifica Trask (TRASK, 2004), a definição de uma unidade sintagmática declara uma gramática sintagmática em que unidades menores (sintagmas) se estruturam para compor frases e sentenças, como vê-se na Figura 2; iii) um dos maiores precursores dessa gramática, Noam Chomsky (CHOMSKY, 1956), conferiu-lhe um aspecto 'algébrico' quando tratada (Veja Tabela 2); iv) sob influência desse linguista, predispôs-se a independência da sintaxe frente à semântica, o que declara a existência de árvores sintáticas coerentes desprovidas de semântica inteligível (CHOMSKY, 2009) e, v) finalmente, nessa teoria foi demonstrada a complexidade da língua humana, por meio do se conhece como Hierarquia de Chomsky que descreve os tipos fundamentais de linguagens formais (CHOMSKY, 1956).

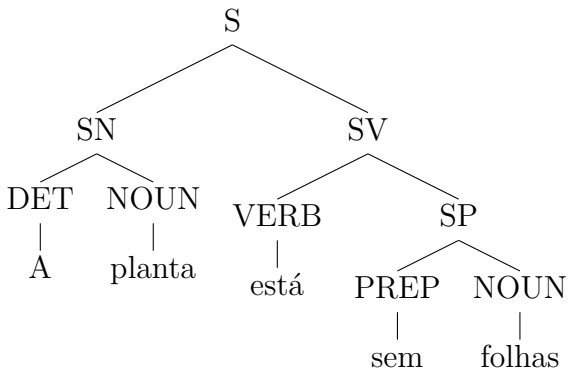

Figura 2 – Árvore de estrutura sintática de constituinte

Para estruturar uma árvore segundo a sintaxe de constituintes, como representamos

na Figura 2, as seguintes regras e léxicos devem ser elaboradas:

Tabela 2 – Regras para composição de constituinte

Regras sintagmáticas		
S	->	SN + SV
SN	->	DET + NOUN
SV	->	VERB + SP
SP	->	PREP + NOUN
Representações léxicas		
DET	->	a
NOUN	->	planta, folhas
VERB	->	está
PREP	->	de

2.6 Sintaxe de dependência

Uma outra alternativa à tradição da gramática de constituintes seria implementar a análise sintática segundo o formalismo de dependência ou gramáticas de dependência, conforme a figura3, fundamentalmente atribuída aos autores Lucien Tesnière (TESNIÈRE, 1959), como também a Louis Hjelmslev (HJELMSLEV, 1975).

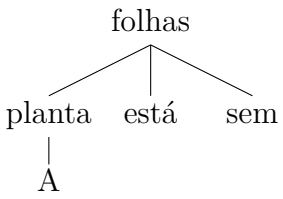

Figura 3 – Árvore de estrutura sintática de dependência (UDPipe)

Numa proposta de Pagani (PAGANI, 2015) se encontra a possibilidade de três tipos de dependência nessa gramática por meio do termo (in(ter))dependência[2]. Assim, dize-se que a sintaxe de dependência implica numa estrutura de dependências (*dependency struture*), e, especificamente, nessa pesquisa, trabalhamos as dependências propostas por Nivre *et al* (NIVRE; MCDONALD, 2008) denominadas de (*Universal Dependency*, UD) a qual escreveremos mais adiante (Seção 2.6.2). Aqui, compete-nos mencionar que esse modelo de anotação tem sua representação na forma de árvores de dependência o que as diferem das árvores de estrutura de constituintes frasais, conforme apresentado na Figura 4.

[2] 'Com o termo (in(ter))dependência, estou abreviando as três possibilidades deste tipo de relação: dependência (não recíproca), independência (mera concatenação, sem dependência de qualquer parte) e interdependência (recíproca).'

Nesta figura, as relações de dependências existentes (***nsubj, det, advmod, obl, case,*** *etc.*) entre os termos da sentença partem de um termo principal denominado ***root*** para seus dependentes, e destes, também, podem partir relações binárias para os que lhes são subordinados, segundo as relações que lhes são próprias frente ao termo principal e/ou aquele que os predicam.

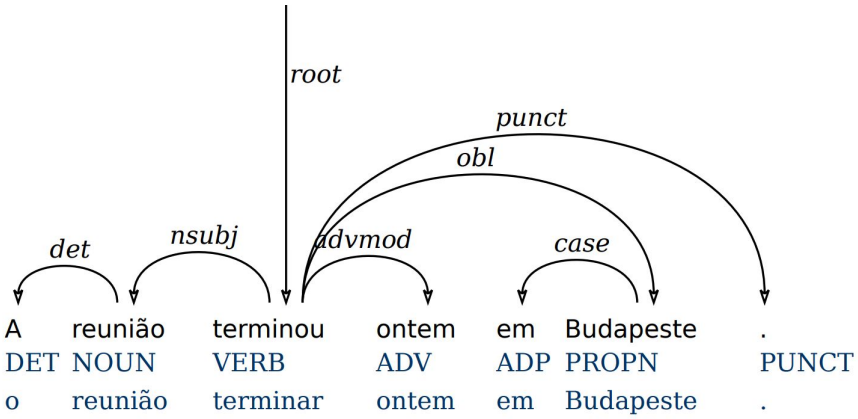

Figura 4 – Árvore de dependência no formalismo UD

2.6.1 Métodos computacionais aplicados a sintaxe de dependência

Ao se discutir acerca dos métodos computacionais que tratam de análise sintática por relações de dependência explorando *treebanks*, os seguintes modelos se destacaram: programação dinâmica, grafos, constraint satisfation, *neural dependency* e *transition-based dependency parsing*.

Programação dinâmica

Richard Bellman (BELLMAN, 1957) definiu como programação dinâmica aqueles métodos implementados com intuição dita 'método tabular' que propõem a resolução de um problema relacionando-o a instâncias mais simples que a original e a cada decisão considera-se todo um novo conjunto de condições antes de seguir à etapa posterior, a intenção aqui é a armazenar resoluções menores numa tabela para recorrer a elas já prontas sempre que precisar, o que pode ser interpretado como uma 'recursão em forma de tabela', uma das aplicações desses algoritmos está relacionada a identificação de termos *heads* por Eisner (HUANG; SAGAE, 2010).

Grafo

É o tratamento onde se considera a relação de um conjunto de elementos chamados nós, ou vértices, por meio de conexões constituídas pelo conjunto das chamadas arestas. Considerando-se, por exemplo, na análise de dependência, os itens lexicais como vértices e as relações que estabelecem uns para com os outros como as arestas, tem-se uma estrutura de dados como recurso. Essa aproximação foi utilizada por MacDonald *et al.* (MCDONALD et al., 2005) para representação da análise de dependência não-projetiva, tanto em línguas ditas *free word order*, como outras.

Constraint satisfation

Essa classe de algoritmos é aplicada a problemas aos quais dada uma série de restrições procura-se o melhor resultado que satisfaça um objetivo, o que os remetem a modelagens que podem ser representadas como que em problemas de programação linear. Karlsson (KARLSSON, 1990) recorreu a essa abordagem para tratamento de tópicos relacionados à ambiguidade, subcategorizados em: pre-processamento, análise morfológica, determinação morfológica local, mapeamento morfossintático, determinação morfológica contextual, definição de limites frasais intra-sentencial e estabelecimento de funções sintáticas.

Neural dependency

Essa aproximação é composta por uma série de algoritmos com sua arquitetura inspirada no mecanismo de funcionamento do tecido neural, em que estruturas menores se conectam formando uma rede que permite ativação de determinadas funções. Atualmente, tanto para PLN (CHEN; MANNING, 2014) como para as mais diferentes áreas e abordagens da computação, esse modelo tem obtido resultados significativos.

Transition-based dependency parsing

Estabelece a aplicação *shift-reduce*, apresentado na Figura 5, é um método considerado determinístico, foi implementado por Nivre (NIVRE; MCDONALD, 2008).

O método usa uma estrutura de pilha para estabelecer as relações existentes entre o termos da sentença. Pode ser descrito da seguinte forma: quando há inserção de uma palavra da sentença na pilha, ocorre o que chama-se *shift* e no momento em que o tipo de relação de dependência que a palavra apresenta é definida com outra palavra ocorre a remoção desta da pilha, chamada *reduce*. Essas operações podem ser exemplificadas pela

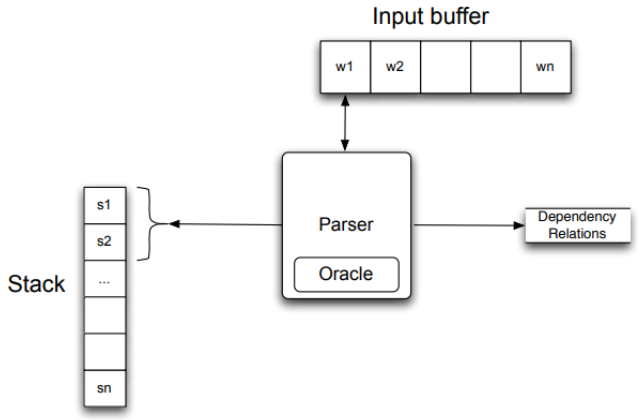

Figura 5 – Intuição *transition-based learning* (JURAFSKY; MARTIN, 2021)

Figura 6, até que todas as relações que dos elementos lexicais existentes na sentença sejam atribuídas.

Step	Stack	Word List	Predicted Action
0	[root]	[book, the, flight, through, houston]	SHIFT
1	[root, book]	[the, flight, through, houston]	SHIFT
2	[root, book, the]	[flight, through, houston]	SHIFT
3	[root, book, the, flight]	[through, houston]	LEFTARC
4	[root, book, flight]	[through, houston]	SHIFT
5	[root, book, flight, through]	[houston]	SHIFT
6	[root, book, flight, through, houston]	[]	LEFTARC
7	[root, book, flight, houston]	[]	RIGHTARC
8	[root, book, flight]	[]	RIGHTARC
9	[root, book]	[]	RIGHTARC
10	[root]	[]	Done

Figura 6 – Demonstração da ação do algoritmo (JURAFSKY; MARTIN, 2021)

2.6.2 O formalismo *Universal Dependency*

Atualmente, outra metodologia de anotação gramatical chamada *Universal dependencies* (UD) (NIVRE; MCDONALD, 2008) está em curso. A UD propõe compor uma descrição de dependência que possa ser usada em qualquer uma das línguas humanas existentes. Essa tendência de anotação, ou formalismo, tem como fundamento estruturas arbóreas sintáticas formadas a partir do registro das relações de dependência entre os elementos formadores de uma sentença. Nessas estruturas de dependência estão ausentes as marcações sintagmáticas (SN, SV, SADJ, etc.) das sentenças, ou seja, elas lhes são implícitas.

De acordo com Marneffe *et al.* (MARNEFFE et al., 2021)), a UD inspirou-se nas pesquisas das anotações da *Stanford Dependencies* (MARNEFFE et al., 2014), *Google Universal Part-of-Speech tags* (PETROV; DAS; MCDONALD, 2011), e da *Interset interlingua for morphosyntactic tagsets* (ZEMAN, 2008).

As anotações de dependência feitas em Stanford por Marneffe *et al.* (MARNEFFE et al., 2014) tinham como pretensão extrair uma taxonomia, uma sistematização, que expressasse as diferentes línguas naturais. Esses estudos permitiram identificar a existência de disposições lexicais comuns relacionadas a sintaxe. Com isso, a proposta de Stanford era a substituição da análise realizada pela gramática de constituintes pela gramática ou formalismo de dependência. Inicialmente a pesquisa dedicava-se unicamente ao problema de implicação textual, em inglês *Recognizing Textual Entailment*, cujo objetivo principal era descrever e aplicar a compreensão semântica existente nas diferentes línguas naturais (POLIAK, 2020).

Já os rótulos UD PoS (*Part-of-speech*) propostos por Petrov *et al.* (PETROV; DAS; MCDONALD, 2011) tiveram como motivação a identificação de similaridade quanto às classes gramaticais das palavras em diversos idiomas. Veja a Tabela 3. Essas classes gramaticais foram obtidas de acordo com suas definições ou seus contextos no discurso, por meio de indução gramatical, também, chamada de inferência gramatical. Comparando experimentalmente vários *treebanks* os autores arquitetaram a universalização das classes gramaticais para a linguística computacional.

Tabela 3 – Rótulos UD PoS (UD, 2021)

Classe aberta	**Classe fechada**	**Outras**
ADJ	ADP	PUNCT
ADV	AUX	SYM
INTJ	CCONJ	X
NOUN	DET	
PROPN	NUM	
VERB	PART	
	PRON	
	SCONJ	

Para completar, as pesquisas de Zeman (ZEMAN, 2008) estabeleceram um meio para extrair anotações morfossintáticas comuns de *corpora* distintos, o que propiciou a interpretação de conteúdos ou atributos característicos dos termos das sentenças. Como consequência dessa corrente teórica, foi apontado pelo autor um incremento no desempenho de *parsers* estatísticos para diversas línguas.

Entre os pesquisadores citados há consenso que o êxito do projeto da UD está ancorado nas seguintes diretrizes: i) promover bases de análise linguística para cada língua; ii) estabelecer uma tipologia que permita paralelismos entre as línguas e famílias de línguas, um desafio se consideradas as descrições anteriores em Greenberg (GREENBERG, 1963), Velupillai (VELUPILLAI, 2012), (DRYER; HASPELMATH, 2013); iii) ser consistente

para um anotador humano; iv) ser compreensível para não especialistas; v) obter alta acurácia em analisadores computacionais e, por fim, vi) atender a tarefas *downstreams* (extração de relação, compreensão de leitura, tradução automática dentre outras) (UD, 2021).

Uma vez apresentado alguns pontos que fundamentam o projeto UD, escrevemos de modo simplificado sobre outra perspectivava que decorre desse estudo, a UD como uma teoria linguística.

2.6.3 UD como teoria linguística

Para se tornar referencial para fins de pesquisa em morfossintaxe, interpretação semântica e em processamento de língua natural, a UD, primariamente, considera que nossas observações do mundo destacam três unidades linguísticas fundamentais: i) os **nomes** que representam as entidades (objetos); ii) as **cláusulas** que descrevem canonicamente eventos (ações ou estados) e; iii) os **modificadores** que atribuem características a nomes, cláusulas e, por vezes, aos próprios modificadores (MARNEFFE et al., 2021).

Pelos princípios da UD, os predicadores, termos que expressam estados ou ações, requerem participantes representados por nomes ou entidades, estabelecendo a relação predicadores–entidades. Sobre esses fundamentos, a UD organiza uma estrutura hierárquica denominada 'gramática de dependência'. Essa gramática constrói uma árvore de relações a partir de um termo definido como *head* para os demais termos chamados de *dependentes*.

As relações de dependência sintática do projeto *Universal Dependencies*, revisadas originalmente das pesquisas de Marneffe (MARNEFFE et al., 2014) estabelece 37 relações de dependência e propõem a descrição de uma taxonomia universal como está disposto na Tabela 4 explicada por Duran (DURAN, 2021).

> Esse quadro destaca os argumentos principais dos predicados, chamados de argumentos *core*, separando-os dos demais argumentos considerados não *core*. Separa, também, os argumentos e modificadores de predicados dos modificadores de nominais. Apresenta, além disso, etiquetas diferentes quando o dependente da relação está sob forma oracional (as quais correspondem às orações subordinadas).

Esse entendimento expressa que as relações de dependência procuram estabelecer, e, ao mesmo tempo, distinguir termos que chamamos **predicadores** daqueles denominados **argumentos**, o que pode se dar de forma recorrente, quando, por exemplo, um termo **argumento** ser o **predicador** de outro elemento. Ainda temos outras situações que consideram questões relacionadas a termos sem dependência direta com o *root* (projetivi-

Tabela 4 – Relações de Dependência Universal

	Nominals	Clauses	Modifier words	Function words
Core arguments	nsubj	csubjt		
	obj	ccomp		
	iobj	xcomp		
Non-core arguments	obl	advcl	advmod	aux
	vocative		discourse	cop
	expl			mark
	dislocated			
Nominal dependents	nmod	acl	amod	det
	apos			clf
	numod			case
Coordination	MWE	Loose	Special	Other
conj	fixed	list	orphan	punct
cc	flat	parataxis	goeswith	root
	compound		reparandum	dep

dade), representados comumente por segmentos subordinados, quanto a isso destaca-se que se permite a composição de subtipos de relações dadas as peculiaridades das línguas expressando 65 de relações distintas, três exemplos dessas são: i) **acl:relcl**, para cláusulas adnominais relativas; ii) **aux:pass**, para o auxiliar passivo e iii) **csubj:outer**, para sujeitos oracionais externos de predicados que são orações, etc.

No capítulo a seguir veremos os principais trabalhos relacionados a esta nossa proposta.

3

Trabalhos relacionados

Ramshaw e Marcus (RAMSHAW; MARCUS, 1995) foram precursores da aplicação de AM quanto a identificação de sintagmas. Eles usaram a metodologia *Transformation-Based Learning*, TBL, para essa função de *shallow parsing*. Este método obteve precisão e revogação na ordem de 92% para SN e 89% para outros sintagmas. A metodologia TBL consiste numa de aprendizagem baseada em transformação que inicia com alguma solução simples (regras iniciais) para identificar o objetivo proposto e, posteriormente, aplica transformações (novas regras) que propõe o incremento do desempenho anterior de marcação dos sintagmas. O TBL é uma abordagem de classificação linear, assim como é o algoritmo de Winnow (LITTLESTONE, 1988), também muito utilizado para *parsing* parcial.

Erik Sang (SANG, 2002), em 2002, para resolver o problema de SP, recorreu à metodologia *memory-based learning* para identificar sintagmas nominais, bem como para o *parsing* completo. Neste artigo, Sang relatou que obteve precisão e revogação de $\approx$93% para os sintagmas nominais.

Molina e Pla (MOLINA; PLA, 2002) aplicaram cadeias escondidas de Markov (HMM) para a mesma atividade. Estes autores conseguiram resultados como F-score $= 93,25\%$, resultados equiparados ao estado da arte para a tarefa de SP na mesma época da pesquisa de Sang.

Choi e colaboradores (CHOI; LIM; CHOI, 2005) trataram o problema de recuperar estruturas sintagmáticas de forma confiável sem utilizar a análise sintática completa e profunda. Os autores encontraram regras gramaticais delimitadoras de sintagmas aplicadas a árvores de decisão, fazendo do método de SP um problema de classificação de forma recursiva. O melhor resultado obtido pelos pesquisadores foi F-score $= 91,7\%$.

Destacamos, ainda, que diferentes línguas naturais foram objeto para emprego da metodologia de SP aliada a métodos de aprendizado de máquina, tal como a língua turca (TOPSAKAL et al., 2017), o hindi-inglês (SHARMA et al., 2016).

Garrido Alenda (ALENDA et al., 2004) lançaram mão do SP para construção de máquinas de tradução português-espanhol e espanhol-português na Universidade de Alicante.

João Ricardo da Silva (SILVA, 2007), em sua tese de doutoramento, trata a segmentação frasal de textos escritos em português usando um SP construídos sobre autômatos de estados finitos. O resultado deste trabalho foi relatado pouco tempo antes (BRANCO; SILVA, 2006) conferindo uma precisão de 99,92% e revocação de 99.95% sobre um *corpus* anotado manualmente de 280 mil *tokens* composto por artigos de jornais e novelas, conhecido como LX-corpus (BRANCO; SILVA, 2004), o que reporta ser o método efetivo de igual forma em dalém das metodologias recorrentes a AM.

Por fim, nossas pesquisas bibliográficas apontam o trabalho de Ophélie Lacroix (LACROIX, 2018) quanto a identificação de marcações sintagmáticas nominais sobre textos escritos em inglês e anotados no formalismo *Universal Dependency* por meio de *Shallow Parsing*.

4

Metodologia

Neste capítulo descrevemos os recursos, as considerações, os passos e os critérios de pesquisa utilizados para a identificação dos sintagmas nominais lexicais. Dividimos este capítulo em duas grandes seções, Dados e Métodos, para facilitar a leitura. Nesta primeira subseção abaixo descrevemos o *corpus* que usamos neste trabalho.

4.1 Dados

Para a identificação de determinadas estruturas da linguagem em PLN, tais como os SN_L, é imprescindível a utilização de um *corpus*. O termo remete a compreensão de um 'conjunto de documentos' que propiciam de si a análise, a exploração e uma eventual comparação dos resultados obtidos entre diversas abordagens de processamento sobre este conjunto de dados. Na literatura relacionam-se muitas definições para esse recurso e, dentre elas, a que pontua com maior clareza acerca da natureza, composição e propósito foi estabelecida por Sánches (SANCHEZ, 1995) melhor traduzido por Sardinha (SARDINHA, 2000) quando diz que um *corpus* consiste em:

> Um conjunto de dados linguísticos (pertencentes ao uso oral ou escrito da língua, ou a ambos), sistematizados segundo determinados critérios, suficientemente extensos em amplitude e profundidade, de maneira que sejam representativos da totalidade do uso linguístico ou de algum de seus âmbitos, dispostos de tal modo que possam ser processados por computador com a finalidade de propiciar resultados vários e úteis para a descrição e análise.

A princípio, tivemos dois pontos a considerar nessa pesquisa para que o método de identificação dos SN_L se tornasse aplicável: o primeiro, o fato da inexistência de um *corpus* o qual haja marcações explícitas de tal estrutura da linguagem escrita por meio de

Relações de Dependência UD; o segundo, a escolha de um *corpus* em língua portuguesa no qual estejam expressas as Relações de Dependência UD.

Como os SN_L têm sua origem nos modelos de anotação de sintagmas frasais, selecionamos o *corpus* Bosque por estar inserido em dois contextos: tanto na gramática de constituintes pelo projeto Linguateca chamado *corpus* Bosque 8.0 (AFONSO et al., 2002) que remete à natureza sintagmática, quanto a versão pertencente ao projeto *Universal Dependencies*, Bosque 2.10 (RADEMAKER et al., 2017), que se relaciona a proposta. Ou seja, para cada sentença nestes dois recursos temos a ocorrência dos SN_L nos dois formalismos, no formalismo de gramática de constituintes e no formalismo UD. Deste modo, pudemos analisar as estruturas sintáticas declaradas por meio da hierarquia dos sintagmas frasais recursivos da primeira gramática para fazermos seleção manual dos tais na segunda que segue a gramática de dependências universais.

Desse modo, as estruturas de árvores deitadas, ou formato ad, conforme Alonso e Freitas (FREITAS; AFONSO, 2008) destacam é documentada de acordo com o modelo expresso pela Figura 7, que permite percorrer seus nós sintagmáticos, bem como explorar suas estruturas internas, tipologia e topologia hierárquica.

```
informação textual
nº frase: texto
A1
NÓ RAIZ<
=NÓ 1
==NÓ 1.1.
===NÓ 1.1.1
====NÓ 1.1.1....n
==NÓ 1.2.
===NÓ 1.2.1.
====NÓ 1.2.1....
```

Figura 7 – Formato ad (FREITAS; AFONSO, 2008)

O *corpus* Bosque é 'composto por 9.367 frases, retiradas os primeiros 1000 extractos (aprox.) dos corpora CETENFolha e CETEMPúblico' que serve como referência para o projeto *Universal Dependencies* proposto, como já mencionado neste trabalho por Nivre e outros (NIVRE; MCDONALD, 2008). *A priori*, o objetivo dessa pesquisa é a investigação de SN_L no português do Brasil e para tal realizamos um recorte de 790 sentenças do *corpus* Bosque no formalismo UD que remetem ao CETENFolha 1.0.

Uma vez estabelecido esse passo, realizamos um recorte de 790 sentenças do *corpus* Bosque, versão 2.10 UD para classificação de seus SN_L. A descrição estatística quanto a quais atributos morfossintáticos UD e relações UD[1], bem como suas frequências nesse

[1] Duas Relações de Dependência UD não ocorrem no Bosque: *classifier* (cls) e *unspecified dependency* (dep).

Tabela 5 – Comparativo Bosque UD frente Recorte SNL

Quantitativo	Bosque UD	Recorte SNL
Sentenças	9357	790
Tokens	210958	16672
Relações de Dependência UD	43	37
UD PoS Tag	17	16

corpus são apresentadas no Anexo A e Anexo B. Sendo esses dois atributos preditivos os elementos recuperados para a obtenção do sintagma de Oliveira e Freitas (OLIVEIRA; FREITAS, 2006). Para representação, na Tabela 5 faz-se uma discreta comparação entre os dois conjuntos.

Entendemos que as diferentes estruturas sentenciais fazem desta proposta uma atividade, por vezes, mais difícil para análise. Nesse contexto, podemos observar no *corpus* a ocorrência de: i) frases – que consistem em todo e qualquer seguimento textual de sentido completo, ii) orações – que compreendem estruturas sintáticas que detém no mínimo um verbo, ora possuindo sujeito e predicado, ora desprovido dum ou outro, nesse caso, reduzindo-se a uma frase e iii) períodos – indicados pela composição de duas ou mais orações. Desse modo, são representadas diferentes construções da língua portuguesa, pois o *corpus*, expressa, mesmo que minimamente, desde sentenças nominais e simples àquelas extensas e mais complexas onde estão presentes fenômenos como: coordenação, elipses, expressões lexicais, etc. nas quais estão marcados os SN_L.

4.1.1 BIO *tags*

Em razão dos SN_L serem uma definição distinta frente aos modelos conceituais de anotação sintagmática existentes, recorremos às marcações IOB *format* (acrônimo para *inside, outside, beginning*) também conhecida por formato BIO introduzidas por Marcus e Ramshaw (RAMSHAW; MARCUS, 1995) em suas pesquisas respectivamente apontam se um *token* está incluso, egresso ou é inicial, dado um segmento alvo.

Esse modelo permite a seleção das estruturas sintagmáticas por meio da marcação de seus fragmentos nas sentenças, aqui postas como instâncias de ocorrência ou não dos SN_L. Assim, numa sentença como a do exemplo 9 apresentada no Capítulo 2 poderíamos expressar as *tags* IOB como na Tabela 6 marcando 'B' como o começo, 'I' como pertencente ou 'O', fora do domínio, de um SN_L.

Tabela 6 – Tabela marcações BIO *format*

token	BIO
Caneta	B
esferográfica	I
Montblanc	I
para	O
escrever	O
em	O
papel	B
apergaminhado	I
de	I
cor	I
sépia	I
.	O

4.1.2 Modelagem da estrutura de dados

O corpus Bosque UD tem sua estrutura de dados definida no formato estabelecido por Buchholz e Erwin (BUCHHOLZ; MARSI, 2006), tidos como grafos hierárquicos acíclicos. Sua arquitetura contém dez campos informacionais distintos para cada palavra/*token*, conforme Figura 8.

ID: Índice de palavras, inteiro começando em 1 para cada nova frase; ou um intervalo para *tokens* de várias palavras; assim como um número decimal para nós vazios (os números decimais podem ser inferiores a 1, mas têm de ser superiores a 0).

FORM: Forma de palavra ou símbolo (sinal) de pontuação.

LEMMA: Lema ou raiz da forma da palavra.

UPOS: Marcação morfossintática de dependência universal.

XPOS: Marcação morfossintática específica do idioma; sublinhar se não estiver disponível[2].

FEATS: Lista de características morfológicas relacionadas pela UD ou de uma extensão específica de linguagem definida; sublinhado se não estiver disponível.

HEAD: Cabeçalho da palavra atual, que é um valor de ID ou zero (0).

DEPREL: Relação de dependência universal com o HEAD (raiz (ou *root*) se HEAD = 0) ou um subtipo específico de idioma definido de um.

[2] Este campo está sublinhado no Bosque

DEPS: Grafo de dependência aprimorado na forma de uma lista de pares *head-deprel*.

MISC: Qualquer outra anotação. ID: Índice da palavra, inteiro começando em 1 para cada nova frase; pode ser um intervalo para *tokens* de várias palavras; pode ser um número decimal para nós vazios (números decimais podem ser menores que 1, mas devem ser maiores que 0). (Tradução Livre)

```
# global.columns = ID FORM LEMMA UPOS XPOS FEATS HEAD DEPREL DEPS MISC
# newdoc_id = CF1
# text = PT no governo
# sent_id = CF1-1
# source = CETENFolha n=1 cad=Opinião sec=opi sem=94a
1       PT      PT      PROPN   B       Gender=Masc|Number=Sing 0       root    _       _
2-3     no      _       _       I       _       _       _       _       _
2       em      em      ADP     I       _       4       case    _       _
3       o       o       DET     I       Definite=Def|Gender=Masc|Number=Sing|PronType=Art       4       det
4       governo governo NOUN    I       Gender=Masc|Number=Sing 1       nmod    _       _

```

Figura 8 – Exemplo da estrutura de arquivos tipo CoNLL-X

Dos quais utilizamos três campos para o algoritmo tradicional (upos, deprel e xpos) e quatro para os demais classificadores (upos, deprel, deps e xpos). Especificando cada um deles, notamos: **upos** – rótulos relativos à morfossintaxe UD (*language-specific part-of-speech tag*), **deprel** – rótulos relacionados às relações de dependência universal, o campo **deps** para lançarmos o atributo morfossintático UD relativo ao *token head* do termo em análise e, finalmente, **xpos** um campo vazio que utilizamos para lançamento dos rótulos IOB-*format*. Com isso, tomamos, dada a estrutura do TBL a combinação inicialmente dos campos **upos** com **xpos** e, posteriormente, dos campos **deprel** com **xpos** para o primeiro e segundo experimento repespectivamente e, por fim, estabelecemos para os demais classificadores os três primeiros campos nos serviram conjuntamente como atributos preditivos (upos, deprel, deps) e o último como atributo alvo (xpos).

A escolha desses campos para identificação de padrões recupera os resultados do conjunto de métodos de *shallow parsing* apontadas no Capítulo 2 de Hammerton *et al.* (HAMMERTON et al., 2002). Recordando em outras palavras, utilizamos dois quocientes distintos, um do método de *chunking* (UD Pos tag) e o outro do método de estabelecimento de relações (Relações de Dependência UD) para obter sintagmas que são decorrentes do *text chunking*.

Após a extração desses dados, compomos uma estrutura *DataFrame* realizando uma apresentação matricial de dados em si mais tratável para os algoritmos de aprendizado de máquina utilizados, semelhante ao apresentado na Tabela 7 (PANDAS, 2021).

Nesse ponto, é interessante observar que nas classificações da sintaxe, UD PoS Tags e Relações de Dependência UD, há um fator sintetizador manifesto, pois dadas as muitas possibilidades de arranjos e construções que os léxicos da língua permitem.

Tabela 7 – Campos com dados de *tokens*.

token	*deprel*	*upos*	*deps*	*iob-format*
PT	root	PROPN	0	B
no			-	I
em	case	ADP	2	I
o	det	DET	3	I
governo	nmod	NOUN	1	I

Estabelecendo uma comparação quantitativa quanto a variedade de rótulos nos atributos das palavras/*tokens* considerando: lema, classe morfossintática e relações de dependências inscritos, nota-se que, dos 16672 *tokens* existentes no *corpus*, foram identificados 3629 lemas e nesses 17 UD PoS Tags e 38 Relações de Dependência UD; o que, consequentemente, tratando dessas duas últimas, reduz a diversidade do conjunto de termos a serem analisados, uma vez que tanto *tokens* como *lemas* remetem a muitas variações; já a morfossintaxe e relações de dependência são representações que reduzem a um conjunto menor de termos distintos.

4.2 Métodos de aprendizado de máquina

Turing (TURING, 2009) estabeleceu a questão seminal ao pontuar se a máquina pode ou não ser capaz de 'pensar'. Essa incipiente interrogação desencadeou uma série de pesquisas norteadas com o objetivo de atribuir às máquinas a faculdade de 'imitar' o comportamento humano segundo Sterrett (MOOR, 2004), o que implica no corolário: pensar – aprender – imitar.

Um desses desdobramentos é definido como aprendizado máquina (AM) – um dos campos da inteligência artificial decorrente de estudos relacionados a teoria do aprendizado computacional e reconhecimento de padrões. O entendimento acerca dessa aproximação é dada por diversos autores na literatura.

Simon (SIMON, 1983) entende que o aprendizado computacional implica num refinamento incremental da precisão de um sistema quanto as suas respostas. Weiss e Kulilowski (WEISS; KULIKOWSKI, 1991) explicam que o aprendizado remete a decisões precisas tomadas pelo sistema computacional com base nas experiências que integram seus exemplos. O que nos aproxima da compreensão de Tom Mitchell (MICHALSKI; CARBONELL; MITCHELL, 2013) que estabelece a seguinte explicação:

> Um programa aprende a partir da experiência E, em relação a uma classe de tarefas T, com medida de desempenho P, se seu desempenho em T, medido por P, melhora com E. (N.T.)

Assim, com a finalidade de abstraírmo-nos do processo de composições de regras sintático-semânticas de modo explícito, delegamos ao computador a tarefa de reconhecimento de padrões que nos levem a identificação mais correta possível da definição dos SN_L; de outra forma, essas regras requereriam ser implementadas manualmente, o que se depararia com questões que envolvem: autômatos de estados finitos, expressões regulares e uma das gramáticas encontradas da Hierarquia de Chomsky, na prática, mais custosas.

Dadas essas considerações, recorremos à utilização de algoritmos de aprendizado máquina supervisionado por reportarem maior fundamentação teórica frente àqueles que se baseiam em aprendizado máquina não supervisionado (MELLO; PONTI, 2018). Dado esse ensejo, passamos a tratar dos algoritmos dos quais um tradicional utilizado na pesquisa de Oliveira e Freitas (OLIVEIRA; FREITAS, 2006) primeiramente explorado e descrito aqui e outros tidos como mais avançados dos quais comentamos posteriormente.

4.2.1 Transform-based learning TBL

O algoritmo de aprendizado supervisionado TBL, implementado por Eric Brill (BRILL, 1995), consiste num modelo de reconhecimento de padrões que permite análises de instâncias que: i) necessitem ser diferenciadas, ii) ou serem identificadas de acordo com um caráter sequencial, ou ainda, iii) apresentem influências contextuais para sua definição (UNESON, 2014). Resumidamente, respectivamente, trata de aspectos de ordem: singular – regida apenas pela ocorrência de uma instância, sequencial – aplicada quando encontra uma sequência regular ou contextual – que compreende termos anteriores e posteriores para identificação.

Ao mesmo tempo, é um código que propicia a apresentação de seus critérios inferenciais, regras, de forma mais explicável se comparado a algoritmos mais avançados. Ele tem como propósito um aprendizado máquina orientado a redução de erros e, por característica, o tratamento de problemas por meio do método chamado guloso, também, chamado míope, descrito por Parberry (PARBERRY, 2021) como segue:

> Um algoritmo guloso propõe a resolução de um subproblema e a aplica e amplia iterativamente até obter a solução do problema maior. Essa solução estende-se de forma 'gananciosa' e considera aspectos locais, sem quaisquer prerrogativas de alcance, a longo prazo, de um ótimo global. (N.T.)

Como tratamos das diferentes composições dos SN_L, de suas formas reduzida – somente baseado num núcleo substantivo, bem como em suas formas mais complexas – compostas de complementos variáveis, julgamos apropriado e plausível a proposta desse algorítimo dada a descrição próxima passada.

De forma complementar, Parberry ressalta que essa classe de algoritmos podem diferir em seus resultados, trazendo tanto os melhores, quanto razoáveis ou resultados ruins; a depender do quão 'gananciosos' serão seus critérios de implementação em face do problema a ser tratado. No caso do TBL, esse ponto está em muito relacionado a um de seus elementos estruturais chamados *templates* que escreveremos mais adiante.

O aprendizado do TBL procura o reconhecimento de padrões por meio da busca de regularidades que melhor descrevem os dados a ele submetidos, elencando as de maior precisão. A composição de suas regras tem por critério um aperfeiçoamento iterativo. Desse modo, uma regra de transformação pode ser reduzida a compreensão de: 'se um contexto (condição) for identificado, é aplicada uma atribuição (classificação)' (N.T.).

Destacamos aqui os elementos imprescindíveis que o algoritmo requer na sua forma estrutural: i) *corpus* – como método de AM supervisionado, o TBL demanda dados anotados como referência para verificação de quão precisas estão suas classificações; ii) *templates* – compreendem os intervalos de domínio fundamentais para identificação de padrões, em outras palavras, amplitude de *tokens* a se considerar para composição de regras e requer conhecimento acerca das proposições e restrições do objeto em análise o que confere ao *template* a posição de 'arquiteto' do método inferencial indutivo.

A configuração de *templates* deve ser considerada fator crítico, acerca desse tema Milidiú (MILIDIÚ; DUARTE; SANTOS, 2007) destaca a importância do parecer de um especialista ou o uso de algoritmos evolutivos para sua construção. iii) fluxo de controle – a) neste modelo, um estado predeterminado de atribuição é aplicado ao texto pré-analisado, conferindo-lhe uma classificação inicial, esta primeira tratativa pode derivar de uma regra extraída do próprio *corpus*, por exemplo, a marcação '**O**' (*outside*) para todos os termos das sentenças, b) agora, classificado, o texto é comparado com o *corpus* de referência, c) os erros identificados são mensurados e utilizados como argumento para nova tratativa, d) considerando os desvios das sucessivas iterações, busca-se uma recomposição do conjunto de regras iniciais que seja mais representativa quanto aos acertos obtidos.

Quanto a isso, pontuamos que diferentes *templates* implicam no que se espera encontrar de padrões nos elementos de uma sentença. A ocorrência de determinada regra, com uma frequência regular e, de forma assertiva, é declarado pelo algoritmo como um padrão representativo, seguido de outros que podem também ser relevantes para a tarefa e àquela vão se agregando. Como resultado desses procedimentos, obtém-se um conjunto de regras que representa da melhor forma possível o aprendizado máquina pretendido. Assim, o 'conhecimento sintático-semântico' é extraído do *corpus* de forma abstrata.

4.2.2 Pontos de destaque do TBL em aplicações sintáticas

Marcus (UNESON, 2014) destaca algumas características relevantes do algoritmo TBL, como: i) interpretabilidade de representação aprendida, ii) síntese da representação apreendida, iii) função objetivo representativa, iv) resistência a *overtraining*, v) pesquisa durante o treinamento em vez de aplicação, vi) integração de recursos heterogêneos e vii) desempenho competitivo.

Notamos ser o TBL um algoritmo voltado para análise de padrões que considera o quesito posicional predominante para analisar os atributos dos termos da sentença relacionados a suas características, ordem de ocorrência e local de ocorrência. Este fato recupera um dos pontos mais relevantes que o PLN trata para definição sintática de um termo ou um segmento: por vezes, não basta elaborarmos regras semelhantes a proposições lógicas, temos de compreender o que um termo é nesta ou naquela posição e, também, o que ele não é, para, assim, nos aproximarmos de um padrão coerente.

Um contraponto a se considerar quanto as correções baseadas na seleção de regras que o algoritmo aplica, vê-se na possibilidade de uma marcação inicialmente correta, ao final do processo, sofrer alterações e findar com uma atribuição incorreta. E mesmo em casos onde os erros não são identificados e continuam nesse estado. A Figura 9 demonstra de forma mais intuitiva os elementos e o fluxo de controle do algoritmo (BRILL, 1995).

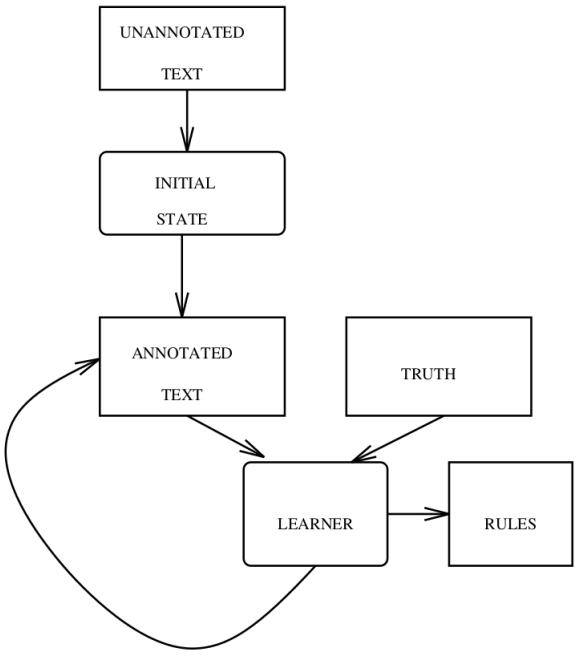

Figura 9 – Aprendizado dirigido por erros

4.2.3 Classificadores

Os algoritmos que atualmente ocupam o estado da arte possuem tratativas que permitem modelos AM analisarem campos vetoriais para a classificação. A proposta desses algoritmos está na busca da redução do viés e da variância dos dados a eles submetidos, ainda mais, são facilmente replicáveis, ou seja, escaláveis para um aprendizado baseado em cooperação, suas configurações permitem tanto o aumento do número de estruturas, bem como diferentes tipos de funções de classificação. trabalhamos aqui com parâmetros padrão.

Árvore de decisão – Os algoritmos inspirados em árvores de decisão tomam como referência principal a apresentação de Quinlan (QUINLAN, 1986). Esse modelo é considerado um dos métodos de modelagem preditiva comumente aplicado em aproximações que incluem questões que envolvem testes de significado estatístico, mineração de dados e aprendizado máquina. Permitem a análise de dados discretos e categóricos (após vetorização), nesses casos, as folhas representam rótulos e os ramos caracterizam as conjunções que levam a rótulos semelhantes (BREIMAN et al., 1984). Representado aqui pelo algoritmo DecisionTreeCalssifier (DTC).

Estendendo-se das árvores de decisão, temos modelos das florestas de decisão que correspondem a seleção aleatória, ou seja, independente e não correlacionadas, de observações para criação de uma árvore de decisão, o que ocorre de forma recorrente, construindo desse modo diversas árvores (floresta) das quais escolhe aquela mais representativa, o que na teoria do aprendizado computacional reflete um princípio de aprendizagem em conjunto (BREIMAN, 2001). Essa implicação remete ao que chamamos de *Bagging* (*Bootstrap Aggregation*) – uma técnica de aprendizado de máquina implementada por Breiman (BREIMAN, 1996) que dispõe de múltiplas instâncias aleatórias do conjunto de dados de treinamento para construir vários modelos de aprendizado e combiná-los para obter uma predição mais precisa.

Em outras palavras, atua construindo diversos modelos de aprendizado usando diferentes amostras aleatórias com substituição do conjunto de dados de treinamento original. Cada modelo, ou *emsemble*, pode ser construído usando um algoritmo de aprendizado diferente ou com diferentes parâmetros do mesmo algoritmo. Em seguida, esses modelos são combinados usando a média (no caso de regressão) ou votação (no caso de classificação) das previsões de cada modelo individual. O objetivo do *bagging* é reduzir a variância dos modelos de aprendizado individual, que é uma medida da sensibilidade do modelo às variações no conjunto de dados de treinamento. Ao construir múltiplos modelos com diferentes amostras aleatórias do conjunto de dados de treinamento, tenta reduzir a sensibilidade dos modelos às variações no conjunto de dados de treinamento, o que pode levar a modelos mais robustos e precisos, nessa pesquisa representados pelo algoritmo RandomForestClassifier (RFC).

O algoritmo *Perceptron* é um modelo linear de AM supervisionado baseado em célula neuronal criado por Rosenblatt (ROSENBLATT, 1958), não é um modelo recente, mas é precursor de outros modelos que se baseiam nas conexões que o tem como unidade, é caracterizado por possuir uma função de ativação entre seus *inputs* e *outputs* sobre os quais atuam a inicialização e a ponderação de pesos que buscam reduzir o diferencial entre saídas calculadas frente as saídas desejadas. Tipicamente é um classificador binário próprio para problemas linearmente separáveis. Sua resposta é dependente se sua função de ativação atinge, ou não, um limiar estabelecido (BISHOP; NASRABADI, 2006).

Perceptron em multicamadas ou *Multilayer Perceptron*, como o nome destaca, é uma rede de aprendizado obtida ao conectarmos diversas camadas de *perceptrons* de modo a formar uma rede de decisões altamente interligada, rede neural. A implementação desse algoritmo permite uma retropropagação de erros (WERBOS, 1990) obtidos por meio de programação direta e, a partir desses, iterativamente, realiza uma série de processamentos reversos. Esses reajustes de elementos de ponderação das funções de ativação existentes, chamados de *[bias]*, têm o propósito de alcançar a maior aproximação possível entre da classificação real e a classificação obtida meio da redução de erros computados por gradiente descendente. Esses algoritmos são tipicamente aplicados a problemas de maior complexidade para identificação de padrões (HINTON, 1990). As técnicas de *deep learning* são tipicamente representadas pela redes neurais.

A inspiração desses modelos podem ser encontrados nas pesquisas de McCullich e Pitts (MCCULLOCH; PITTS, 1943) descrevem um aprendizado de máquina baseado em redes neurais artificiais é chamado aprendizado profundo – um subconjunto do aprendizado de máquina que envolve o treinamento de redes neurais artificiais profundas para identificar padrões complexos nos dados. Uma rede neural artificial é uma rede de nós interconectados, que são semelhantes aos neurônios do cérebro humano. Cada nó recebe entradas de outros nós e produz uma saída, que é então enviada para outros nós. As redes neurais são compostas por várias camadas de nós interconectados, cada uma responsável por extrair características dos dados em um nível diferente de abstração. Esses algoritmos têm alcançado destaque nas áreas de processamento de linguagem natural, sua capacidade de aprender representações hierárquicas de dados, permitindo que eles extraiam características complexas e não-lineares dos dados de entrada lhe conferem atualmente destaque, sendo nessa aproximação demonstrado pelo algoritmo MLPClassifier (MLPC).

Gradiente descendente ponderado em árvores e florestas de decisão – são algoritmos baseados em árvores se florestas de decisão com conceito de árvore escalável na qual o crescimento está fundamentado em computação sequencial e gradientes de regressão ponderados para a classificação, dentre suas características destacam-se: i) a regularização e a ii) poda de árvore para evitar *overfitting*; iii) a amostragem de linha e coluna para melhorar a eficiência computacional; iv) os pesos em árvores para lidar com conjuntos

de dados desbalanceados e o v) processamento paralelo para acelerar o treinamento do modelo. Aponta-se neles os seguintes modos Ridge e Lasso de regularização de variáveis preditivas para o tratamento de *overfitting*, que consiste no ajustamento dos pesos dos recursos ou variáveis independentes para reduzir a complexidade.

A regularização Lasso adiciona uma penalidade à soma dos valores absolutos dos pesos dos *inputs*, levando a muitos pesos desses recursos se aproximarem de zero, descartando aqueles irrelevantes ao modelo. Assim, a regularização Lasso atua quando há muitos recursos em um modelo, e nem todos são relevantes para a predição. A regularização Ridge, por outro lado, acresce uma penalidade à soma dos quadrados dos pesos dos recursos, induzindo a pesos mais uniformes em todos os recursos. O que evita a amplificação de pequenos ruídos nos dados e melhora a generalização do modelo. A regularização Ridge é empregada quando todos os recursos do modelo são relevantes e importantes para a predição. Ambas as técnicas de regularização podem ser usadas juntas como uma combinação chamada *Elastic Net* (ZOU; HASTIE, 2005) que combina as penalidades Lasso e Ridge para fornecer um equilíbrio entre a seleção de recursos e a estabilidade da predição (CHEN; GUESTRIN, 2016).

Essas regularizações são aplicadas sobre um aprendizado obtido por meio do que nomeamos *Boosting*, uma técnica de aprendizado de máquina arquitetado por Schapire (SCHAPIRE, 1990). Essa abordagem combina múltiplos modelos de aprendizado de maneira sequencial, onde cada modelo é construído com base nos erros do modelo anterior. O objetivo do *boosting* é refinar a precisão e o desempenho do modelo de aprendizado ao reduzir o viés do modelo. Assim, diferentemente do *bagging*, que constrói modelos de aprendizado independentes, o *boosting* constrói modelos de aprendizado em sequência, onde cada modelo é construído com base nos erros do modelo anterior. A cada iteração, o modelo tenta corrigir os erros do modelo anterior, dando mais peso aos exemplos que foram classificados incorretamente pelo modelo anterior. Dessa forma, o modelo se concentra em exemplos que são mais difíceis de classificar e tenta reduzir o erro nos exemplos mais difíceis; aqui representados pelos algoritmos XGBClassifier (XGBC) e XGBRFClassifier (XGBRFC).

4.3 Métricas

Para se estabelecer meios para ponderar a atuação em AM diferentes medidas de AM foram criadas, em razão disso, passamos a mencionar de início sua natureza e posteriormente suas interpretações (BISHOP; NASRABADI, 2006).

A natureza das métricas em aprendizado de máquina está no que chamamos matriz de confusão, essa matriz objetiva descrever o desempenho de um modelo. Ela

apresenta a frequência com que cada classe do conjunto de dados foi classificada correta ou incorretamente pelo *ensemble*. A matriz compõem o quadro de valores possíveis diante de um problema de classificação: i) Verdadeiro Positivo (VP): ocorre quando o modelo classifica corretamente uma instância como positiva; ii) Falso Positivo (FP): ocorre quando o modelo classifica incorretamente uma instância como positiva; iii) Verdadeiro Negativo (VN): ocorre quando o modelo classifica corretamente uma instância como negativa e iv) Falso Negativo (FN): ocorre quando o modelo classifica incorretamente uma instância como negativa. Uma matriz de confusão é pautada em tabela de dimensão quadrática, ou m x m, número de linhas igual ao número de colunas, onde as linhas representam as classes reais e as colunas representam as classes preditas pelo modelo:

Tabela 8 – Matriz de confusão

	Predito Positivo	Predito Negativo
Real Positivo	VP	FN
Real Negativo	FP	VN

A partir da matriz de confusão é possível calcular métricas de desempenho do classificador como: precisão, *recall*, Medida-F e acurácia.

Precisão é uma métrica de desempenho usada em problemas de classificação que mede a proporção de instâncias positivas classificadas corretamente pelo modelo. Em outras palavras, a precisão mede a qualidade das previsões positivas do modelo. A precisão é calculada pela seguinte fórmula:

$$\text{Precisão} = \frac{(\text{VP})}{(\text{VP} + \text{FP})} \tag{4.1}$$

Essa medida varia de 0 a 1, onde 1 representa um modelo perfeito que nunca faz predições positivas incorretas e 0 representa um modelo que não acerta nenhuma predição positiva. Sua indicação está relacionada a questões onde falsos positivos implicam em decisões críticas, isso significa que mensurar algo com alta confiança pode implicar em agravo na decisão, pois uma dada instância nem sempre pode corresponder as características esperadas, é uma reserva quanto a confiança absoluta. Porém, é sensível a distribuições de classes não está desequilibrada ou quanto o custo de falsos negativos é importante. De forma pueril, poderíamos resumi-la em: não posso confiar plenamente, pois pode haver coisas erradas nesse conjunto, ela é penalizada pela detecção de FP.

Recall, também conhecido como sensibilidade, é uma métrica de desempenho usada em problemas de classificação que mede a proporção de instâncias positivas reais que foram corretamente identificadas pelo modelo. Em outras palavras, o *recall* mede a qualidade

das predições positivas do modelo em relação ao número total de instâncias positivas no conjunto de dados.

$$\textit{Recall} = \frac{(\mathsf{VP})}{(\mathsf{VP} + \mathsf{FN})} \tag{4.2}$$

Da mesma forma, o *recall* tem seu valor entre o intervalo de 0 a 1, onde 1 representa um modelo perfeito que nunca deixa de identificar instâncias positivas e 0 representa um modelo que não consegue identificar nenhuma instância positiva. O *recall* é uma métrica útil quando o custo de falsos negativos é alto, explicando de outra forma, indica que há o que poderíamos chamar de hiperfoco acerca do aprendizado. Deixando de observar variações de características nas variáveis preditivas que circunscrevem o objeto como verdadeiro positivo. Para definir com a mesma intenção interpretativa que a anterior, poderíamos dizer acerca do conjunto apontado por essa medida: ainda há elementos que estão fora daqui e não deveriam estar, assim é comprometida pela alta de FN.

Já a medida-F é uma métrica de desempenho usada em problemas de classificação que combina a precisão e o *recall* em uma única medida. A medida-F é uma média de natureza harmônica entre a precisão e o *recall*, é calculado pela seguinte fórmula:

$$\textit{Medida-F} = 2 * \frac{(\mathsf{Precisão} * \mathsf{Recall})}{(\mathsf{Precisão} + \mathsf{Recall})} \tag{4.3}$$

A medida-F é útil quando se deseja encontrar um equilíbrio entre a precisão e o *recall*, especialmente quando as classes são desequilibradas, o que leva seu intervalo que está entre 0 e 1 a apontar valores mais baixos. A medida-F é uma métrica útil para avaliar o desempenho do modelo de forma equilibrada. No entanto, assim como a precisão e o *recall*, a medida-F pode ser enganoso em algumas situações, como quando as classes têm diferentes custos de erro. Por isso, é importante escolher a métrica apropriada para o problema em questão e avaliar o desempenho do modelo usando várias métricas relevantes. Resgatando um princípio de rigor matemático-estatístico, podemos dizer que a média de natureza harmônica é menor ou igual à medida de natureza geométrica que é igual ou menor que a média aritmética, portanto mais rigorosa.

Por fim, a acurácia, uma métrica de desempenho usada em problemas de classificação que mede a proporção de instâncias corretamente classificadas pelo modelo em relação ao número total de instâncias no conjunto de dados. Em outras palavras, a acurácia mede a qualidade geral das predições do modelo. Seu cálculo é feito do seguinte modo:

$$\mathsf{Acurácia} = \frac{(\mathsf{VP} + \mathsf{VN})}{(\mathsf{VP} + \mathsf{VN} + \mathsf{FP} + \mathsf{FN})} \tag{4.4}$$

A acurácia é uma métrica simples de interpretar, e é útil quando as classes são equilibradas e o custo de falsos positivos e falsos negativos é semelhante. No entanto, a

acurácia pode ser enganosa quando as classes são desequilibradas ou quando o custo de falsos positivos e falsos negativos é diferente.

No próximo capítulo abordaremos os resultados dos experimentos.

5

Resultados

Neste capítulo apresentamos os resultados obtidos na pesquisa com os diferentes algoritmos utilizados e algumas considerações.

Acerca da natureza e formação do *corpus*, tomamos as primeiras 790 sentenças do *corpus* Bosque UD e analisamos frente as suas pares contidas nos extratos Bosque Linguateca por constituinte, CETENFolha, que por estarem documentadas em formato ad, veja capítulo 4, seção 4.1, predispôs explorarmos a hierarquia e estrutura interna dos SN considerando as especificações de Oliveira e Freitas (OLIVEIRA; FREITAS, 2006) quanto as composições do SN_L. Onde, uma vez, identificada a estrutura correspondente a um SN_L eram lançadas manualmente no *corpus* Recorte SN_L[1] as *tags* BIO-*format* explicitando, assim, nesse pequeno *corpus*, as marcações sintagmáticas do objeto em estudo, descrito na tabela 9, marcações UD PoS Tag e Relações de Dependência UD. Podemos entender que esse recente formalismo gramatical, UD, traz uma perspectiva a ser explorada pela nomeação e identificação das relações de dependência existentes entre os termos de uma sentença, pois as arestas hierárquicas são nomeadas e declaradas de forma explícita a todos os termos analisados, o que permite submetê-los a identificação de padrões, a primeira vista, não percebidos, conforme os resultados que serão apresentados nas seções 5.1 e 5.2.

Tabela 9 – Tabela *corpus* Recorte SNL

Sentenças	790
Tokens	16672
Campos CoNLL	10
Relações UD	38
UD PoS Tag	16
Medida de dados	1020 kB

[1] O campo XPOS agora contém marcações BIO-*tags*

Com relação à seleção dos conjuntos de treino e teste para submissão aos algoritmos de AM, temos duas formas distintas de apreciação que se dão em razão de suas diferentes propostas de aproximação, o que nos permitiu nomeá-los nesse trabalho como algoritmo tradicional, o TBL, e, por um contrafeito, algoritmos avançados, aos demais utilizados.

Para o primeiro, dizemos que selecionamos os encadeamentos de dados de (aprox.) 553 sentenças do nosso *corpus* para treino e de 237 das sentenças restantes para teste, todas armazenadas em lista de *strings*. A estrutura para análise que o algoritmo TBL requer são díades de *strings* como *inputs* e a sequência de apresentação dessas dá-se como na leitura humana, da esquerda para direita de cima para baixo, o que respeita o encadeamento lógico da sintaxe na escrita, preservando o padrão do português, nesse caso.

Os demais algoritmos receberam seus *inputs* codificados em vetores, munidos com o mesmo conteúdo elementar, esses *tokens* vetoriais foram armazenados em estrutura tabular denominada DataFrame, de onde são selecionados de forma aleatória pelo método *train-test-split* contido em uma das bibliotecas da *Application Programming Interface*, ou API, *scikit-learn*, dedicada a ciência de dados e, com efeito, falamos aqui de aproximadamente 11740 instâncias para aprendizado e 5032 para teste, separadas nas proporções 70/30 para cada fase respectivamente.

Desta forma, o *corpus* é analisado em busca de padrões e aqui vemos, ao menos num primeiro momento, que o encadeamento sintático natural é fracionado de forma arbitrária devido ao uso de funções nativas de bibliotecas, *train_test_split*, e, posteriormente, caso o volume de dados seja grande, padrões são identificados e utilizados. De maneira que, por considerar o *corpus* pequeno em termos de PLN da atualidade, remetê-lo ao método de validação cruzada seria, ao nosso ver, penalizá-lo duas vezes. Esse conflito se dá de um lado por estarmos no ambiente sintático e, de outro, por esses algoritmos serem estruturados para essa apresentação. Motivação que nos remeteu a experimentos com o TBL.

5.1 Métricas do algoritmo TBL

O algoritmo TBL com sua estrutura natural de *inputs* permite a entrada de uma única categoria de dados preditivos, assim lançamos num primeiro momento, as Relações de Dependência UD para o treinamento desse classificador e, *a posteriori*, os atributos relativos a UD PoS Tag, ambas sempre associadas as marcações BIO. Para o primeiro recurso, o percentual de $87,42\%$ foi alcançado, abaixo do obtido com o uso das UD PoS Tags que atingiram $91,44\%$ de acurácia, se mostrando mais representativas e com padrão mais regular pelo uso de apenas dois *templates* com os quais compuseram 6 regras conforme exposto na Tabela 10.

Tabela 10 – Tabela resultados finais

	Templates	Regras	Acurácia
Relações de Dependência UD	8	57	87,42%
UD PoS Tag	2	6	91,44%

5.1.1 Regras do TBL

Neste ponto, procuramos interpretar as regras, padrões, identificadas pelo algoritmo TBL tanto com o uso das marcações Relações de Dependência UD, como das marcações UD PoS Tag.

Para as marcações de Relações de Dependência UD, nos deparamos com uma maior diversidade de *templates*, total de 8 *templates*, bem como uma maior composição de regras, somando 57 regras, o que é parcialmente demonstrado na Tabela 11, esse fato explica-se pela existência de uma ampla quantidade de marcações se comparada a morfossintaxe UD, o que permite a conjugação mais vasta de eventos nos quais existam segmentos identificados como SN_L.

Tabela 11 – Principais regras compostas pelas Relações de Dependência UD e rótulos BIO.

Template	*tag* Inicial	*tag* Final	Regra
017	'B'	'I'	(*token*[-1], 'det'), (*token*[1], 'flat:name')
009	'B'	'O'	(*token*[-1], 'nsubj')
017	'B'	'I'	(*token*[-1], 'case'), (*token*[1], 'flat:name')
000	'I'	'B'	(*tag*[-1], 'O')
010	'B'	'O'	(*token* [1]), 'acl:relcl')
017	'I'	'B'	(*token*[-1], 'root'), (*token*[1]),'obj')

Algumas explicações acerca das estatísticas que se referem quando utilizadas as Relações de Dependência UD apontam a ocorrência de 1859 erros inicialmente identificados e após o treinamento a redução desse número para 1545 erros, alcançando uma acurácia de 87,42%, ou $+2,55 p.p.$.

Há uma ampla variedade de *templates* utilizadas, ora estabelecendo padrões por meio das Relações de Dependência UD, ora mediante aos próprios rótulos BIO. Podemos observar que há uma diversificação quanto a inserção ou exclusão das inscrições BIO, não se estabelecendo um tipo inicial prevalecente de alguma delas antes e após o treinamento.

O algoritmo ainda nos traz o número de *templates* que utilizou, 18 desses; e o número de combinações que formulou, 173 criadas. O *template* mais pontuado representou 51,90%

de todos eles e foi responsável por arquitetar 32 regras diferentes, ou seja tríades onde as palavra/*tokens* anterior e posterior se alteravam para identificação de uma sequência correspondente a um SN_L.

Já para as marcas feitas recorrendo as UD PoS Tag, percebemos uma alta representatividade dos SN_L, com a identificação de um número discreto tanto de *templates*, apenas 2, quanto de regras, somente 6 regras, indicado pela Tabela 12. O algoritmo ainda reporta, nesse experimento, a identificação de 553 sequências, esses dados significam que essas marcações alcançaram alta desempenho para as diferentes assinaturas lexicais do SN_L.

Tabela 12 – Principais regras compostas pelas UD PoS Tag e rótulos BIO.

Template	*tag* Inicial	*tag* Final	Regra
017	'O'	'B'	(*token*[-1], 'ADP'), (*token*[1], 'NOUN')
017	'O'	'I'	(*token*[-1], 'DET'), (*token*[1], 'NOUN')
017	'O'	'B'	(*token*[-1], 'ADP'), (*token*[1], 'PROPN')
017	'O'	'I'	(*token*[-1], 'NUM'), (*token*[1], 'NOUN')
001	'O'	'B'	(*tag* [1]), 'I')
017	'O'	'B'	(*token*[-1], 'ADP'), (*token*[1]),'SCONJ')

Ao interpretarmos outros dados estatísticos do algoritmo notamos que foram analisados 12.283 *tokens* no total, sendo que na fase inicial, o modelo apresentava 1.088 *tokens* incorretamente etiquetados, o que corresponde a uma taxa de acerto de 91,14%. Após o treinamento, o modelo apresentou um total de 1.051 *tokens* incorretamente etiquetados, o que corresponde a uma taxa de acerto de 91,44%.

As principais regras estão identificadas por um número (017 ou 001), há predomínio de alterações de rótulos de inscrição inicial ('O'), para rótulos de inscrição final ('B' ou 'I') e um padrão composto por um ou mais elementos. Cada elemento é uma tupla que consiste em uma palavra em uma posição específica (por exemplo, a palavra na posição -1) e um rótulo de parte do discurso (por exemplo, 'NOUN' para um substantivo). O algoritmo ainda descreve que foi utilizado uma estrutura de *templates* com um tamanho de 92. Nesse caso específico, o *template* com identificador 017 obteve pontuação de 91,9% se comparado aos demais, indicando que foi considerada um dos mais importantes pelo modelo, pois compôs 5 regras que representaram cerca de 83,3% do total de regras criadas, isso indica que o padrão identificado é uma palavra/*token* com um determinado valor atribuído à sua posição anterior e outro valor atribuído à sua posição posterior. Já o *template* com identificador 001 obteve uma pontuação de 0,81%, representando uma importância menor em relação à outra, pois estabeleceu apenas uma regra, representando cerca de 16,7% do total de regras formadas, esse modelo diz que o padrão imediatamente anterior identificado

é uma palavra/*token* com um determinado valor atribuído à sua posição.

Nesta tabela 12 destacamos que pelo *template* sequencial, os *tokens* imediatamente anterior e imediatamente posterior ao de análise, a composição de regra em que uma marcação 'ADP' inicial e uma marcação 'NOUN' final delimitavam o segmento correspondente a um SNL, o que se repetiu aos pares: 'DET' – 'NOUN', 'ADP' – 'PROPN', 'NUM' – 'NOUN', essas marcações compuseram as regras mais representativas para identificação do sintagma em análise. De semelhante forma, as demais métricas utilizadas: precisão, revocação e medida-F, também pontuam a valor do uso UD PoS Tag conforme vê-se na Tabela 13 na qual a revocação para classificação da marcação 'B', inicio de um SN_L reportou $93,81\%$.

Tabela 13 – Resultados dos demais algoritmos

Rótulos	Métricas utilizadas			
	Precisão	*Revocação*	*Medida-F*	*Acurácia*
Relações de Dependência UD				87,42
tag B	78,66	88,02	83,08	–
tag I	88,79	76,34	82,09	–
tag O	**88,94**	**89,47**	**89,20**	–
UD PoS Tag				**91,44**
tag B	**92,18**	**93,81**	**92,99**	–
tag I	90,91	85,89	88,33	–
tag O	91,05	92,91	91,97	–

Estes resultados ampliaram os testes iniciais com *corpus* com menor quantidade de exemplares, 101 sentenças, o que está exposto no Anexo C, além disso, nota-se que o algoritmo TBL com o recurso UD PoS Tag se mostrou mais assertivo em $+4,44$ *p.p.* de acurácia quando munido de mais exemplos em suas bases, ainda nesse contexto, sua acurácia demonstra êxito de $+4,81$ *p.p.* frente ao algoritmo MLPClassifier, baseado em modelo conexionista, que obteve o segundo melhor resultado dados os demais algoritmos aplicados.

Ressaltamos nesse ponto, que o algoritmo TBL quanto a composição de regras com o uso de marcações de Relações UD é caracterizado, ao menos inicialmente, por uma correção mais variada, aplicando *tags* BIO para correção dos erros encontrados, como vê-se na tabela 11, porém, ao ser munido de marcações UD PoS Tags é reconhecido por inserir termos, inicialmente, fora do domínio do SN_L, como termos iniciais ou inclusos no segmento de interesse, ver campos *tag* Inicial e *tag* Final na tabela 12.

Em decorrência disso, concluímos que o custo computacional com o uso de Relações

de Dependência UD tanto para escolha dos *templates* como para a composição de regras assumem uma ordem de complexidade próxima à exponencial se comparadas ao treino somente com as marcações UD PoS Tags, em outras palavras, para os *templates* salta de 2 para 8 *templates*; e para formação das regras, estende-se de 6 para 57 regras, além de ocorrer um decremento da acurácia com essa ampliação de complexidade.

5.2 Métricas dos algoritmos classificadores

Tratando-se dessa classe de algoritmos, cabe destacar que os dados que nos servem de recursos derivam de *corpora* e, diferentemente dados numéricos, têm em si estruturas implícitas de ordem tipológica. Todos eles, permitem a associação dos atributos preditivos para busca de padrão, ou seja, nesse momento, temos o pareamento dos dados UD PoS Tags, juntamente a Relações de Dependência UD para determinação se um termo corresponde a qual *tag* BIO-*format*, o que representou um ganho importante de desempenho se tomados os preditores isoladamente, como tornam evidentes as tabelas do anexo D. Isto posto, exploramos algoritmos que utilizam as seguintes técnicas de AM: *bagging*, *boosting* e *deep learning*.

Os modelos de aprendizado de máquina baseados em árvores de decisão são tradicionalmente admitidos sob a perspectiva sintática em PLN. Nessa pesquisa, um modelo elementar dessa classe de algoritmos foi aplicado, DTC – árvore, com intento de apontar quão expressivo é diante sua modelagem replicada de aprendizado por cooperação, RFC – floresta, representante clássico da técnica de *bagging*; com isso, obtivemos para o primeiro uma medida-F de 86,48%, apenas $-0,18$ *p.p.* inferior às florestas de decisão, em outras palavras, esse diferencial seria pouco expressivo dada a complexidade computacional demandada. O destaque quanto as métricas de AM está na precisão alcançada, 87,06%, o que significa que tem uma boa generalização dos dados corretamente classificados frente aos incorretamente classificados.

Diferentemente dos algoritmos anteriores, DTC e RFC, os algoritmos XGBC e XGBRF carregam a técnica de *boosting* aliadas as características declaradas nos seus exemplares anteriores. Nesse cenário, o algoritmo baseado de árvore de decisão com *boosting* atingiu o percentual de precisão de 87,19%, que representa o melhor desempenho que todos os demais algoritmos de AM nomeados como avançados nessa pesquisa.

As redes neurais multicamadas retratada pelo algoritmo MLPC consegui a segunda posição nessa seleção, ficando atrás, somente do algoritmo XGBC. O que nos foi mais significativo nessas comparações está no incremento de aprendizado que expressam frente a sua configuração elementar, PCPT, em $+3,52$*p.p.* de acurácia.

De forma analítica, esses resultados obtidos pelos algoritmos avançados levantam

outra questão de pesquisa, em razão do TBL possivelmente, ser sensível a questões de tipologia, seus resultados podem ser inferiores se comparados com esses se aplicado em outros idiomas de tipologia diversa, o que enseja um ponto motivacional para estudos em outras línguas. Esses aprendizados têm como recurso apenas 1020 Kilobytes de dados, um volume inexpressivo para o atual estado da arte que trabalha com Terabytes de volumes de dados.

Tabela 14 – Resultados obtidos com classificadores baseados em árvore e florestas de decisão com *boosting*

Classificador	Métricas utilizadas			
	Precisão	*Revocação*	*Medida-F*	*Acurácia*
DecisionTreeClassifier	86,91	86,35	86,48	86,35
RandomForestClassifier	87,06	86,53	86,66	86,53
Perceptron	84,98	83,11	83,57	83,11
MLPClassifier	87,18	86,63	86,76	86,63
XGBClassifier	**87,19**	**86,65**	**86,78**	**86,65**
XGBRFClassifier	86,33	86,07	86,16	86,07

A respeito das BIO-*tag*, o formato BIO *tag* se mostrou mais apropriado para identificar SN_L por tratar sequências de coordenação de elementos, assim, onde ocorre sujeito composto (Pedro, Lucas e Carlos são...) seria possível discriminar cada um dos seus entes, seguindo a definição das autoras naturais, o que não seria possível se considerássemos somente *tags* IO (*inside – outside*).

Logicamente questões que envolvem interpretações distintas nos dois formalismos gramaticais em sentenças extensas dificultaram algumas decisões. O termo tomado como vértice, ou nó, em uma gramática não ocupa necessariamente o mesmo *status* noutra para a mesma sentença analisada. Do mesmo modo, algumas marcações remetiam a ponderação do que a definição que as autoras Oliveira e Freitas (OLIVEIRA; FREITAS, 2006) estabelecem, ora a composição da estrutura interna do segmento, ora as implicações naturais da relação predicador/argumento existentes entre seus termos, e isso reafirma o posicionamento de Elhadad (ELHADAD, 1996), escrita no capítulo 2, página 38, onde temos que a estrutura dos SN é hierárquica e mais complexa que a estrutura sentencial ou predicado/argumento.

Referenciando a tipologia da língua portuguesa, tomando os estudos apontados no seção 2.4, página 41, vê-se que ela apresenta, na disposição básica de seus constituintes, a ordenação SVO. Entende-se que a disposição dos elementos frasais expressam as categorias sintáticas de sujeito, verbo e objeto. Essa ordenação significa que ao falarmos ou escrever um texto na língua portuguesa, encontrar-se-á como estrutura regular ocorrências onde o

primeiro constituinte será o sujeito e seus termos, seguido do verbo e seus termos, e em último, o objeto e seus termos.

No tocante aos apostos, ao se considerar as entidades nomeadas nessa pesquisa, uma especificação distinta foi encontrada para os SN_L. Observamos em casos onde a entidade nomeada pode ser interpretada como núcleo do SN_L, como complemento ou objeto. E há ocorrências em que são marcadas como aposto na gramática UD. Na literatura encontramos uma variedade de apostos: explicativo, enumerativo, distributivo, circunstancial e especificativo (FERREIRA, 2018). Quanto a esse último, consideramos fazer ele parte do segmento sintagmático por considerar a relação hierárquica semanticamente mais apropriada que a relação predicado/argumento importante, veja o caso, exemplo 35:

35. O rio **Amazonas** deságua no Atlântico. (FERREIRA, 2018).

Ainda nesse contexto, a interpretação estrutural de casos 'uma avenida **paulista**' e 'um embaixador **sul americano**' são aproximações que procuramos seguir.

Outros casos estão relacionados a expressões adverbiais que internamente podem possuir palavras morfossintaticamente classificadas como substantivos, mas numa interpretação do seu contexto sintático, exercem a funções de locuções adverbiais, o que veementemente as afastam do propósito de identificação e classificação da pesquisa, são exemplos delas: **por exemplo, a nível de, na medida em que**, etc., onde os termos 'exemplo', 'nível' e 'medida' são substantivos, porém nesse contexto de UD, optamos por excluí-los. Para esses pontos, a análise de um especialista em linguística seria interessante a fim de contribuir para maior e melhor expressividade de confiança do *baseline* construído.

6

Conclusão

Primeiramente, quanto ao contexto da pesquisa, resgatamos a definição de Hammerton, no capitulo 2, página 27 que remete o SP a um conjunto de métodos aplicados que procura determinar: a morfossintaxe, os sintagmas ou as relações de dependência existentes em segmentos textuais. Procuramos, aqui, recuperar um conteúdo informacional, chamado de SN_L, delineado no capítulo 2, página 35, por meio de morfossintaxe UD (UD PoS Tag) e Relações de Dependência UD. Esse sintagma, bem como outros, está explicitamente ausente na sintaxe da *Universal Dependencies* (NIVRE; MCDONALD, 2008), tratado, então, com um método específico de *shallow parsing* chamado *text chunk*. Recorrendo a esse método computacional, lançamos mãos a técnicas de AM descritas no capítulo 4, na seção 4.2, para exploramos, por meio de abstração de regras, o *corpus* Bosque, estudado nessa pesquisa, em duas naturezas ou versões distintas, por constituintes e por dependências universais, tendo, a rigor, dois *corpora* distintos.

A análise sintática parcial se mostrou uma estratégia factível para identificação de SN_L por meio de técnicas de aprendizado de máquina submetidos esse recente formalismo de dependências universais, tanto com uso de algoritmo tradicional, como também com algoritmos nomeados mais avançados.

A inserção de *tags* BIO no *corpus* criado demonstrou expressar os SN_L dentro das definições de Oliveira e Freitas (OLIVEIRA; FREITAS, 2006). Se considerados os ensaios preliminares tabela 18 e o experimento final tabela 13, notamos que o algoritmo TBL é representativo quanto a definição das BIO *tags* por meio da UD PoS Tags.

Podemos dizer que o SN_L é um quociente lexical de natureza nominal, sintática e amplo quanto ao seu gradiente de complexidade e cardinalidade, o que lhe confere diferentes assinaturas lexicais.

Quanto ao tratamento computacional podem ser extraídos por meio das marcações UD PoS Tags e Relações de Dependência UD. O algoritmo TBL destacou-se com o percentual de $+4,02$ *p.p.* com a morfossintaxe UD em acurácia se comparadas as marcações de dependência UD. E frente aos demais algoritmos obteve uma métrica de acurácia, no

mínimo, de $+4,79$ *p.p.*.

As pesquisas realizadas por Oliveira e Freitas inseridas em outro contexto sintático, constituintes, alcançaram uma precisão de 85,9% e uma medida-F de 86,2%, estabelecer um comparativo entre essa e aquela pesquisa dados seus diferentes contextos seria inadequado, as proximidade entre as métricas confirmam a afirmação de Rambow (RAMBOW, 2010) que as gramáticas de constituintes e de dependência trazem o mesmo conteúdo sintático sob diferentes perspectivas.

A morfossintaxe UD aliada as relações de dependência UD são elementos que permitem estabelecer segmentos sintagmáticos não naturais da gramática de dependências universais.

Pontuamos que a dependência de muitos dos algoritmos atuais quanto ao volume e variedade de dados para extração de padrões pode ter influência nos resultados, veja anexo D frente aos resultados anotados na tabela 14, tal sujeição é questionada por Santos (SANTOS, 2021) quanto os atuais critérios de implementação de algoritmos e reafirmaram a escolha do TBL como referência.

O desempenho do TBL pelos ensaios no idioma português do Brasil não representa um resultado apropriado para afirmações que remetam a um AM aplicável a todas as línguas naturais por meio do formalismo UD, pois o algoritmo TBL quando estabelece uma sequência para identificação do $\mathrm{SN_L}$ está subordinado a tipologia dessa língua em específico. Lembrando que a proposta UD é estabelecer uma gramática comum todas as línguas naturais conhecidas.

O aprendizado computacional que recorre à classificação de rótulos BIO permite a identificação de fragmentos que compõe um $\mathrm{SN_L}$ e, com isso, suas configurações mais extensas podem ter seus limites mal definidos ou descontinuados.

Dentre as aproximações feitas com algoritmos avançados, a técnica de *boosting* se destacou ligeiramente as técnicas de *deep learning*. Há uma homogeneidade entre resultados da tabela 14 apresentando-se muito próximos em termos percentuais dadas as métricas tomadas.

6.1 Contribuições futuras

Por fim, destacamos como contribuições futuras: i) a ampliação do *corpus* com maior quantidade de sentenças anotadas para reafirmar ou não do desempenho do TBL frente aos tais algoritmos nomeados como mais avançados; ii) a revisão do *corpus* por linguistas; iii) aproximações que incrementem a precisão e a revocação alcançadas até este momento e iv) a identificação desse tipo específico de sintagma em outros idiomas para reafirmar a

proposta do projeto *Universal Dependencies*, bem como a correlação dos SN_L nas diferentes línguas naturais e v) verificar se com o uso restrito a Relações UD em outra língua natural a questão tipológica pode ser perpassada.

Referências

ABNEY, S. P. Parsing by Chunks. In: _____. *Principle-Based Parsing: Computation and Psycholinguistics*. Dordrecht: Springer Netherlands, 1992. p. 257–278. ISBN 978-94-011-3474-3. Disponível em: <https://doi.org/10.1007/978-94-011-3474-3_10>.

ADAM, J.-M. *Les textes: types et prototypes: récit, description, argumentation, explication et dialogue*. [S.l.]: Armand Colin, 2011.

AFONSO, S. et al. Floresta sintá(c)tica: A treebank for Portuguese. In: *Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC)*. Las Palmas, Spain: [s.n.], 2002. p. 1698–1703. Disponível em: <http://www.lrec-conf.org/proceedings/lrec2002/sumarios/1.htm>.

AIRES, R. V. X. *Implementação, adaptação, combinação e avaliaçao de etiquetadores para o português do Brasil*. Tese (Doutorado) — Universidade de São Paulo, 2000.

ALENDA, A. G. et al. Shallow parsing for Portuguese-Spanish machine translation. In: *Workshop Notes of TASHA'2003*. Lisboa, Portugal: Edições Colibri, 2004. p. 21–24. Disponível em: <http://hdl.handle.net/10045/27523>.

BEIJSTERVELDT, L. M. van; HELL, J. van. Lexical noun phrases in texts written by deaf children and adults with different proficiency levels in sign language. *International Journal of Bilingual Education and Bilingualism*, Taylor & Francis, v. 13, n. 4, p. 439–468, 2010.

BELLMAN, R. A Markovian Decision Process. *Journal of Mathematics and Mechanics*, Indiana University Mathematics Department, v. 6, n. 5, p. 679–684, 1957. ISSN 00959057, 19435274. Disponível em: <http://www.jstor.org/stable/24900506>.

BICK, E. Automatic parsing of Portuguese. In: *Proc. Second Workshop on Computational Processing of Written Portuguese (Curitiba, 23-25 October 1996)*. [S.l.: s.n.], 1996. p. 91–100.

BISHOP, C. M.; NASRABADI, N. M. *Pattern recognition and machine learning*. [S.l.]: Springer, 2006. v. 4.

BRANCO, A.; SILVA, J. Evaluating Solutions for the Rapid Development of State-of-the-Art POS Taggers for Portuguese. In: *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*. Lisbon, Portugal: European Language Resources Association (ELRA), 2004. Disponível em: <https://www.aclweb.org/anthology/L04-1000>.

BRANCO, A. et al. The CINTIL and LX Companion Collections of Language Resources and Tools for Portuguese. *Proceedings, ToRPorEsp*, 2014.

BRANCO, A.; SILVA, J. R. A suite of shallow processing tools for Portuguese: LX-suite. In: *Proceedings of the Eleventh Conference of the European Chapter of the Association for Computational Linguistics: Posters & Demonstrations*. Trento, Italy: Association for Computational Linguistics, 2006. p. 179–182. Disponível em: <https://www.aclweb.org/anthology/E06-2024.pdf>.

BREIMAN, L. Bagging predictors. *Machine learning*, Springer, v. 24, p. 123–140, 1996.

BREIMAN, L. Random forests. *Machine learning*, Springer, v. 45, n. 1, p. 5–32, 2001.

BREIMAN, L. et al. Classification and regression trees. Belmont, CA: Wadsworth. *International Group*, v. 432, p. 151–166, 1984.

BRILL, E. Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part-of-Speech Tagging. *Comput. Linguist.*, MIT Press, Cambridge, MA, USA, v. 21, n. 4, p. 543–565, dez. 1995. ISSN 0891-2017.

BUCHHOLZ, S.; MARSI, E. CoNLL-X Shared Task on Multilingual Dependency Parsing. In: *Proceedings of the Tenth Conference on Computational Natural Language Learning*. USA: Association for Computational Linguistics, 2006. (CoNLL-X '06), p. 149–164.

CHEN, D.; MANNING, C. D. A fast and accurate dependency parser using neural networks. In: *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. [S.l.: s.n.], 2014. p. 740–750.

CHEN, T.; GUESTRIN, C. XGBoost: A Scalable Tree Boosting System. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: Association for Computing Machinery, 2016. (KDD '16), p. 785–794. ISBN 9781450342322. Disponível em: <https://doi.org/10.1145/2939672.2939785>.

CHOI, M.-S.; LIM, C. S.; CHOI, K.-S. Automatic Partial Parsing Rule Acquisition Using Decision Tree Induction. In: DALE, R. et al. (Ed.). *Natural Language Processing – IJCNLP 2005*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005. p. 143–154. ISBN 978-3-540-31724-1.

CHOMSKY, N. Three models for the description of language. *IRE Transactions on Information Theory*, v. 2, n. 3, p. 113–124, 1956.

CHOMSKY, N. *Syntactic structures*. [S.l.]: De Gruyter Mouton, 2009.

CHURCH, K.; PATIL, R. Coping with Syntactic Ambiguity or How to Put the Block in the Box on the Table. *American Journal of Computational Linguistics*, v. 8, n. 3-4, p. 139–149, 1982. Disponível em: <https://aclanthology.org/J82-3004>.

COWIE, J.; LEHNERT, W. Information Extraction. *Commun. ACM*, Association for Computing Machinery, New York, NY, USA, v. 39, n. 1, p. 80–91, jan. 1996. ISSN 0001-0782. Disponível em: <https://doi.org/10.1145/234173.234209>.

DRYER, M. S.; HASPELMATH, M. (Ed.). *WALS Online*. Leipzig: Max Planck Institute for Evolutionary Anthropology, 2013. Disponível em: <https://wals.info/>.

DURAN, M. S. *Manual de anotação de relações de dependência*. [S.l.]: ICMC, 2021. 79 p.

ELHADAD, M. Lexical choice for complex noun phrases: Structure, modifiers, and determiners. *Machine Translation*, v. 11, p. 159–184, 03 1996.

FERRARI, A. T. *Metodologia da Pesquisa Científica*. [S.l.]: McGRAW-HILL, 1982.

FERREIRA, N. O. APOSTO EXPLICATIVO: UMA PROPOSTA DE ANÁLISE À LUZ DA SEMÂNTICA COGNITIVA. *Anais do VIII SAPPIL-Estudos de Linguagem*, 2018.

FREITAS, C.; AFONSO, S. *Bíblia Florestal: Um manual lingüístico da Floresta Sintá(c)tica*. Linguateca, 2008. Disponível em: <https://www.linguateca.pt/Floresta/BibliaFlorestal/completa.html>.

GIVÓN, T. *Syntax: an introduction. Volume II*. [S.l.]: John Benjamins, 2001.

GODBY, C. J. *A computational study of lexicalized noun phrases in English*. [S.l.]: The Ohio State University, 2002.

GRAHL, J. A. P. *Estrutura de Constituintes*. Dissertação (Mestrado) — UFPR, 2009. Disponível em: <https://docs.ufpr.br/~arthur/orients/joao_inic.pdf>.

GREENBERG, J. Some Universals of Grammar with Particular Reference to the Order of Meaningful Elements. In: . [S.l.: s.n.], 1963.

HAMMERTON, J. et al. Introduction to Special Issue on Machine Learning Approaches to Shallow Parsing. *Journal of Machine Learning Research*, JMLR, v. 2, p. 551–558, mar 2002. ISSN 1532-4435.

HINTON, G. E. Connectionist learning procedures. In: *Machine learning*. [S.l.]: Elsevier, 1990. p. 555–610.

HJELMSLEV, L. *Prolegômenos a uma teoria da linguagem*. [S.l.]: Perspectiva, 1975.

HUANG, L.; SAGAE, K. Dynamic programming for linear-time incremental parsing. In: *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. [S.l.: s.n.], 2010. p. 1077–1086.

JURAFSKY, D.; MARTIN, J. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. [s.n.], 2021. v. 3. 158 p. Disponível em: <https://web.stanford.edu/~jurafsky/slp3/>.

KARLSSON, F. Constraint grammar as a framework for parsing running text. In: *COLING 1990 Volume 3: Papers presented to the 13th International Conference on Computational Linguistics*. [S.l.: s.n.], 1990.

KITCHENHAM, B. Procedures for performing systematic reviews. *Keele, UK, Keele University*, v. 33, n. 2004, p. 1–26, 2004.

KRUG, T. C.; MERGEN, S. L. S. Extração de Relacionamentos a Partir de Sites da Wikipedia. *Anais do Salão Internacional de Ensino, Pesquisa e Extensão*, v. 4, n. 2, mar. 2013. Disponível em: <https://periodicos.unipampa.edu.br/index.php/SIEPE/article/view/60025>.

KURAMOTO, H. Uma abordagem alternativa para o tratamento e a recuperação de informação textual: os sintagmas nominais. IBICT, 1996.

KUSHMERIK, N. Gleaning the Web. *IEEE Intelligent Systems and their Applications*, v. 14, n. 2, p. 20–22, 1999.

LACROIX, O. Investigating NP-Chunking with Universal Dependencies for English. In: *Proceedings of the Second Workshop on Universal Dependencies (UDW 2018)*. Brussels, Belgium: Association for Computational Linguistics, 2018. p. 85–90. Disponível em: <https://aclanthology.org/W18-6010>.

LI, X.; ROTH, D. Exploring evidence for shallow parsing. In: *Proceedings of the 2001 Workshop on Computational Natural Language Learning*. USA: Association for Computational Linguistics, 2001. (ConLL '01, v. 7). Disponível em: <https://doi.org/10.3115/1117822.1117826>.

LITTLESTONE, N. Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine Learning*, Springer, v. 2, n. 4, p. 285–318, 1988.

MAIA, L. C.; SOUZA, R. R. Uso de sintagmas nominaikuras na classificação automática de documentos eletrônicos. *Perspectivas em Ciência da informação*, SciELO Brasil, v. 15, p. 154–172, 2010.

MARNEFFE, M.-C. D. et al. Universal Stanford Dependencies: A cross-linguistic typology. In: *Proceedings of the Nineth International Conference on Language sources and Evaluation (LREC'14)*. Reykjavik, Iceland: European Language Resources Association (ELRA), 2014. p. 4585–4592. Disponível em: <http://www.lrec-conf.org/proceedings/lrec2014/pdf/1062_Paper.pdf>.

MARNEFFE, M.-C. de et al. Universal Dependencies. *Computational Linguistics*, v. 47, n. 2, p. 255–308, 07 2021. ISSN 0891-2017. Disponível em: <https://doi.org/10.1162/coli_a_00402>.

MARTINS, R. T.; HASEGAWA, R.; NUNES, M. d. G. V. Curupira: um parser funcional para a língua portuguesa. *Relatório Técnico. São Carlos: NILC*, 2002.

MCCULLOCH, W. S.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, Springer, v. 5, p. 115–133, 1943.

MCDONALD, R. et al. Non-projective dependency parsing using spanning tree algorithms. In: *Proceedings of human language technology conference and conference on empirical methods in natural language processing*. [S.l.: s.n.], 2005. p. 523–530.

MELLO, R. F.; PONTI, M. A. *Machine Learning: A Practical Approach on the Statistical Learning*. Springer International Publishing, 2018. Disponível em: <http://dx.doi.org/10.1007/978-3-319-94989-5>.

MICHALSKI, R. S.; CARBONELL, J. G.; MITCHELL, T. M. *Machine learning: An artificial intelligence approach*. [S.l.]: Springer Science & Business Media, 2013.

MILIDIÚ, R. L.; DUARTE, J. C.; SANTOS, C. N. d. Evolutionary TBL template generation. *Journal of the Brazilian Computer Society*, Springer, v. 13, n. 4, p. 39–50, 2007.

MOLINA, A.; PLA, F. Shallow Parsing using Specialized HMMs. *Journal of Machine Learning Research*, Microtome Publishing, v. 2, n. 4, p. 595–613, 2002.

MOOERS, C. N. Zatocoding applied to mechanical organization of knowledge. *American Documentation*, Wiley Online Library, v. 2, n. 1, p. 20–32, 1951.

MOOR, J. H. The Turing Test: The Elusive Standard of Artificial Intelligence. *Computational Linguistics*, v. 30, p. 115–116, 2004.

NEVES, M. H. de M. *Gramática de usos do português*. [S.l.]: Unesp, 2000.

NIVRE, J.; MCDONALD, R. Integrating graph-based and transition-based dependency parsers. In: *Proceedings of ACL-08: HLT*. [S.l.: s.n.], 2008. p. 950–958.

OLIVEIRA, C.; FREITAS, M. C. d. *Um modelo de sintagma nominal lexical na recuperação de informações*. [S.l.]: sn, 2006. 778–786 p.

OTHERO, G. d. A. *Grammar Play: um parser sintático em Prolog para a língua portuguesa*. Tese (Doutorado) — Pontifícia Universidade Católica do Rio Grande do Sul, 2004.

PAGANI, L. A. *Duas Noções Fundamentais para Gramáticas de Dependência*. 2015. Disponível em: <https://docs.ufpr.br/>.

PANDAS, P. *Dataframe*. [S.l.]: Boxplot—Pandas, 2021.

PARBERRY, I. *Problems on algorithms*. [S.l.: s.n.], 2021.

PARTEE, B. H.; MEULEN, A. t.; WALL, R. E. Mathematical Methods in Linguistics. *Canadian Journal of Linguistics/Revue canadienne de linguistique*, Cambridge University Press, v. 39, n. 1, p. 50–52, 1994.

PETROV, S.; DAS, D.; MCDONALD, R. T. A Universal Part-of-Speech Tagset. *CoRR*, abs/1104.2086, 2011. Disponível em: <http://arxiv.org/abs/1104.2086>.

PHRIDVIRAJ, M. S. B.; RAO, C. V. G. An Approach for Clustering Text Data Streams Using K-Means and Ternary Feature Vector Based Similarity Measure. In: *Proceedings of the The International Conference on Engineering & MIS 2015*. New York, NY, USA: Association for Computing Machinery, 2015. (ICEMIS '15). ISBN 9781450334181. Disponível em: <https://doi.org/10.1145/2832987.2833081>.

POLIAK, A. *A Survey on Recognizing Textual Entailment as an NLP Evaluation*. 2020.

QUINLAN, J. R. Induction of decision trees. *Machine learning*, Springer, v. 1, p. 81–106, 1986.

RADEMAKER, A. et al. Universal Dependencies for Portuguese. In: *Proceedings of the Fourth International Conference on Dependency Linguistics (Depling 2017)*. Pisa,Italy: Linköping University Electronic Press, 2017. p. 197–206. Disponível em: <https://aclanthology.org/W17-6523>.

RADFORD, A. *Syntactic Theory and the Structure of English: A Minimalist Approach*. [S.l.]: Cambridge Textbooks in Linguistics, 1981.

RAMBOW, O. The Simple Truth about Dependency and Phrase Structure Representations: An Opinion Piece. In: *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Los Angeles, California: Association for Computational Linguistics, 2010. p. 337–340. Disponível em: <https://aclanthology.org/N10-1049>.

RAMSDEN, M. J. *Introduction to Index Language Construction*. [S.l.]: Clive Bingley, 1974.

RAMSHAW, L.; MARCUS, M. Text Chunking Using Transformation-Based Learning. *Third ACL Workshop on Very Large Corpora. MIT*, Springer, p. 157–176, 12 2002.

RAMSHAW, L. A.; MARCUS, M. P. Text Chunking using Transformation-Based Learning. *CoRR*, cmp-lg/9505040, 1995. Disponível em: <http://arxiv.org/abs/cmp-lg/9505040>.

RINO, L. H. M.; PARDO, T. A. S. A Sumarização Automática de textos: principais características e metodologias. In: *Anais do XXIII Congresso da Sociedade Brasileira de Computação*. [S.l.: s.n.], 2003. v. 8, p. 203–245.

ROSENBLATT, F. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, American Psychological Association, v. 65, n. 6, p. 386, 1958.

SANCHEZ, A. *Cumbre: corpus lingüístico del español contemporáneo: fundamentos, metodología y aplicaciones*. [S.l.]: Sociedad general espanola de librería, 1995.

SANG, E. F. T. K.; BUCHHOLZ, S. Introduction to the CoNLL-2000 Shared Task Chunking. In: *Fourth Conference on Computational Natural Language Learning and the Second Learning Language in Logic Workshop*. [s.n.], 2000. Disponível em: <https://aclanthology.org/W00-0726>.

SANG, E. T. K. Memory-Based Shallow Parsing. *Journal of Machine Learning Research*, Microtome Publishing, v. 2, p. 559–595, 2002.

SANTOS, D. S. M. Grandes quantidades de informação: um olhar crítico. In: *II Congresso Internacional em Humanidades Digitais*. Online: UFRJ, 2021. Disponível em: <https://youtu.be/Qi-3QzP0NxM>.

SARDINHA, T. B. Linguística de Corpus: histórico e problemática. *DELTA: Documentação de Estudos em Lingüística Teórica e Aplicada*, v. 16, 01 2000.

SCHAPIRE, R. E. The strength of weak learnability. *Machine learning*, Springer, v. 5, p. 197–227, 1990.

SHARMA, A. et al. Shallow Parsing Pipeline – Hindi-English Code-Mixed Social Media Text. In: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. San Diego, California: Association for Computational Linguistics, 2016. p. 1340–1345. Disponível em: <https://aclanthology.org/N16-1159>.

SILVA, J. R. M. F. da. *Shallow processing of Portuguese: From sentence chunking to nominal lemmatization*. Tese (Doutorado) — Universidade de Lisboa, Faculdade de Ciências, 2007.

SILVA, M. C.; KOCH, I. G. *Linguística aplicada ao português.* . [S.l.]: Cortez, 2012.

SILVA, V. L. P. P. D. Sintagmas nominais complexos: critérios formais e funcionais de identificação, com reflexos na construção do gênero acadêmico. *Revista Linguística*, v. 16, n. Esp., p. 666–679, 2020. ISSN 2238-975X. Disponível em: <https://revistas.ufrj.br/index.php/rl/article/view/43728>.

SIMON, H. A. 2 - WHY SHOULD MACHINES LEARN? In: MICHALSKI, R. S.; CARBONELL, J. G.; MITCHELL, T. M. (Ed.). *Machine Learning*. San Francisco (CA): Morgan Kaufmann, 1983. p. 25–37. ISBN 978-0-08-051054-5. Disponível em: <https://www.sciencedirect.com/science/article/pii/B9780080510545500066>.

SVOBODOVÁ, I. *Sintaxe da língua Portuguesa*. Brno: Masarykova Univerzita, 2014. ISSN 978-80-210-7026-4.

TESNIÈRE, L. *Elements of structural syntax*. Paris: C. Klincksieck, 1959.

TOPSAKAL, O. et al. Shallow parsing in Turkish. In: *2017 International Conference on Computer Science and Engineering (UBMK)*. [S.l.: s.n.], 2017. p. 480–485.

TRASK, R. L. *Dicionário de Linguagem e Linguística*. [S.l.]: Contexto, 2004. 368 p. Tradução e adaptação de Rodolfo Ilari. Revisão Técnica de Ingedore Villaça Koch e Thaís Cristófaro Silva. ISBN 85-7244-254-5.

TURING, A. M. Computing machinery and intelligence. In: *Parsing the turing test*. [S.l.]: Springer, 2009. p. 23–65.

UD. *Short Introduction to UD*. 2021. 27, de agosto de 2021. Disponível em: <https://universaldependencies.org/>.

UNESON, M. When errors become the rule: Twenty years with transformation-based learning. *ACM Comput. Surv.*, Association for Computing Machinery, New York, NY, USA, v. 46, n. 4, apr 2014. ISSN 0360-0300. Disponível em: <https://doi.org/10.1145/2534189>.

VELUPILLAI, V. *An introduction to linguistic typology*. [S.l.]: John Benjamins Publishing, 2012.

WEISS, S. M.; KULIKOWSKI, C. A. *Computer Systems That Learn: Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning, and Expert Systems*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1991. ISBN 1558600655.

WERBOS, P. J. Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, IEEE, v. 78, n. 10, p. 1550–1560, 1990.

WU, Y.; ZHAO, J.; XU, B. Chinese Named Entity Recognition Combining a Statistical Model with Human Knowledge. In: *Proceedings of the ACL 2003 Workshop on Multilingual and Mixed-Language Named Entity Recognition - Volume 15*. USA: Association for Computational Linguistics, 2003. (MultiNER '03), p. 65–72. Disponível em: <https://doi.org/10.3115/1119384.1119393>.

ZEMAN, D. Reusable Tagset Conversion Using Tagset Drivers. In: *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*. Marrakech, Morocco: European Language Resources Association (ELRA), 2008. Disponível em: <http://www.lrec-conf.org/proceedings/lrec2008/pdf/66_paper.pdf>.

ZHAI, C.; MASSUNG, S. *Text Data Management and Analysis: A Practical Introduction to Information Retrieval and Text Mining*. [S.l.]: Association for Computing Machinery and Morgan & Claypool, 2016. ISBN 9781970001174.

ZOU, H.; HASTIE, T. Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, Wiley Online Library, v. 67, n. 2, p. 301–320, 2005.

A

Descrição UD PoS Tag do *corpus* Recorte SNL

Tabela 15 – Descrição da classificação morfossintática e ocorrência

Morfossintaxe	Frequência
PROPN	1339
None	1039
ADP	2177
DET	2394
NOUN	2960
VERB	1487
ADV	560
ADJ	750
PUNCT	2096
NUM	298
SYM	55
PRON	453
AUX	366
CCONJ	329
SCONJ	347
X	16
INTJ	6

B

Descrição de Relações UD do *corpus* Recorte SNL

Tabela 16 – Relações UD e número de ocorrências

Relação UD	Frequência	Relação UD	Frequência
root	790	obl	705
None	1039	cop	204
case	2120	cc	336
det	2379	conj	387
nmod	1172	mark	353
parataxis	121	expl	44
nsubj	845	xcomp	219
flat:name	425	aux	82
acl	157	nsubj:pass	67
advmod	530	aux:pass	79
obj	759	obl:agent	43
amod	627	iobj	41
punct	2096	csubj	30
advcl	176	compound	15
nummod	184	flat	32
appos	229	discourse	3
acl:relcl	162	flat:foreign	8
ccomp	104	vocative	1
fixed	107	list	1

C

Resultados preliminares TBL

Tabela 17 – Tabela resultados preliminares.

	Templates	Regras	Acurácia
Relações UD	8	12	80,9%
UD PoS Tag	4	8	82,7%

Tabela 18 – Métricas percentuais preliminares.

Rótulos	Métricas utilizadas			
	Precisão	*Revocação*	*Medida-F*	*Acurácia*
Relações UD				85,1
tag B	74,6	66,6	70,4	–
tag I	83,7	76,1	79,7	–
tag O	80,8	88,1	84,3	–
UD PoS Tag				87,0
tag B	77,3	69,0	72,9	–
tag I	86,2	79,3	82,6	–
tag O	82,0	88,8	85,3	–

D

Resultados preliminares algoritmos

Tabela 19 – Desempenho preliminar com *inputs* em separado

Algoritmo	Acc. Relações UD	Acc. UD Pos Tag
DecisionTreeClassifier	72,65	76,41
RandomForestClassifier	72,65	76,41
Perceptron	52,82	65,47
MLPClassifier	72,65	76,41
XGBClassifier	71,62	76,41
XGBRFClassifier	64,96	73,33

Tabela 20 – Desempenho preliminar com *inputs* em conjunto Relações UD + UD PoS Tag

Algoritmo	Precisão	*Recall*	*F-mensure*	Acurácia
DecisionTreeClassifier	80,14	80,00	79,92	80,00
RandomForestClassifier	80,13	80,00	79,90	80,00
Perceptron	80,97	78,46	78,62	78,46
MLPClassifier	79,83	79,66	79,53	79,66
XGBClassifier	80,28	80,00	79,94	80,00
XGBRFClassifier	80,72	80,85	80,75	80,85