Programa de Pós-Graduação em

Computação Aplicada

Mestrado Acadêmico

Augusto Lopes da Silva

THOTH: Um Algoritmo para Geração de Frases Curtas em Linguagem Natural a partir de Dados Abertos e Conectados

São Leopoldo, 2019

AUGUSTO LOPES DA SILVA

THOTH: UM ALGORITMO PARA GERAÇÃO DE FRASES CURTAS EM LINGUAGEM NATURAL A PARTIR DE DADOS ABERTOS E CONECTADOS

Dissertação apresentada como requisito parcial para obtenção do título de Mestre em Computação Aplicada, pelo Programa de Pós-Graduação em Computação Aplicada da Universidade do Vale do Rio dos Sinos – UNISINOS.

Orientador: Prof. Dr. Sandro José Rigo.

São Leopoldo

2019

S586t Silva, Augusto Lopes da.
Thoth : um algoritmo para geração de frases curtas em linguagem natural a partir de dados abertos e conectados / Augusto Lopes da Silva. – 2019.
134 f. : il. ; 30 cm.

Dissertação (mestre) – Universidade do Vale do Rio dos Sinos, Programa de Pós-Graduação em Computação Aplicada, 2019.
"Orientador: Prof. Dr. Sandro José Rigo."

1. Dados Abertos e Conectados. 2. Geração de Linguagem Natural. 3. RDF. I. Título.

CDU 004

Dados Internacionais de Catalogação na Publicação (CIP)
(Bibliotecária: Amanda Schuster – CRB 10/2517)

Augusto Lopes da Silva

THOTH: Um Algoritmo para Geração de Frases Curtas em Linguagem Natural a partir de Dados Abertos e Conectados

Dissertação apresentada à Universidade do Vale do Rio dos Sinos – Unisinos, como requisito parcial para obtenção do título de Mestre em Computação Aplicada.

Aprovado em 28 de março de 2019.

BANCA EXAMINADORA

Prof. Dr. Sandro José Rigo – UNISINOS.

Prof.ª Dra. Patrícia Jaques Maillard – UNISINOS

Prof.ª Dra. Carla Amor Divino Moreira Delgado – UFRJ

Prof. Dr. Sandro José Rigo (Orientador)

Visto e permitida a impressão
São Leopoldo,

Prof. Dr. Rodrigo da Rosa Righi
Coordenador PPG em Computação Aplicada

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior Brasil (CAPES) - Código de Financiamento 001

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001

AGRADECIMENTOS

Este trabalho teve apoio de diversas pessoas para que pudesse ser desenvolvido. Primeiramente, é dever agradecer à minha família por todo o suporte fornecido durante o desenvolvimento desta dissertação. Agradecer minha noiva que, durante a escrita desta dissertação, se tornou minha esposa e ajudou no desenvolvimento desta pesquisa.

Agradecer meu orientador, professor Dr. Sandro José Rigo, por todo seu apoio e orientação durante o desenvolvimento deste trabalho. Agradecer os linguistas Jéssica Braun de Moraes, Filipe Ramos e a professora Dra. Isa Mara Alves no embasamento linguístico necessário para esta dissertação. Também é meu dever agradecer a Letícia Menegotto no apoio para construção da avaliação estatística das avaliações.

Além disso, meus sinceros agradecimentos aos avaliadores: professores Michael, Melissa, Marie, Lori, Barbra, Voigt, Roberto, Carol e Vera pelo tempo dedicado à avaliação das sentenças geradas pelo algoritmo.

RESUMO

A atual consolidação e disponibilização de bases de dados abertos e conectados vem fomentando diversas iniciativas, sendo que, dentre elas, observa-se o uso do conteúdo armazenado para geração de linguagem natural. A geração de frases em linguagem natural pode ser beneficiada com o uso destas bases em pelo menos dois aspectos, que são a grande quantidade de informações disponível e a existência de anotações adicionais sobre o significado destas informações.

Quanto aos recursos usados para a lexicalização das frases, os trabalhos nesta área podem ser agrupados em três categorias, sendo a primeira caracterizada pela utilização de *templates* para a definição da estrutura das frases, a segunda pelo uso de algoritmos de aprendizado de máquina para a geração das frases de modo não supervisionado e a terceira a utilização de ambas abordagens em um modelo híbrido.

As abordagens geram resultados considerados interessantes, porém apresentam dificuldades em relação à naturalidade das sentenças geradas. Observa-se que os trabalhos relacionados ao tema não utilizam em ampla escala as informações das propriedades RDF presentes nas ontologias, fatores que podem ser considerados como apoio na geração de frases mais naturais. Dentre essas informações estão relacionamentos semânticos entre conceitos que podem ajudar a construção de sentenças em linguagem natural. Diante deste contexto, a pesquisa atual visa explorar essas propriedades para geração de linguagem natural para o idioma inglês a partir de um conjunto de *templates* elaborados por linguistas e do uso de recursos lexicais. Foram executados duas avaliações para ajustar critérios e variáveis para o algoritmo de geração de linguagem proposto e um terceiro para validação final da pesquisa. A primeira avaliação buscou identificar formas de geração de frases em linguagem natural a partir das propriedades RDF. Partindo da análise dos resultados da primeira avaliação, uma nova avaliação foi conduzida buscando medir a naturalidade das sentenças geradas a partir das propriedades RDF.

Por fim, uma terceira avaliação foi projetada e executada, onde profissionais da linguística e nativos do idioma inglês avaliaram as frases curtas geradas pelo algoritmo. Os resultados da avaliação final foram considerados promissores para aplicações que objetivem geração de linguagem natural a partir das informações das propriedades RDF com apoio de recursos lexicais.

Palavras-Chave: Dados Abertos e Conectados. Geração de Linguagem Natural. RDF.

ABSTRACT

The current consolidation and availability of linked open data have fomented several initiatives, among them it is possible to observe the use of the content stored in them for natural language generation. The generation of natural language phrases can benefit from using these bases in at least two aspects, which are the large amount of information available and the existence of additional notes on the meaning of this information.

As for the resources used for the lexicalization of sentences, the works in this area can be grouped into three categories: the first one characterized by the use of sets of templates to define the sentence structure; the second by the use of machine learning algorithms to the generation of sentences in an unsupervised way; and the third the use of both approaches in a hybrid model.

The approaches generate interesting results but have difficulties in relation to the naturalness of the sentences generated. It is observed that the works related to the topic do not use on a large scale the information of the RDF properties present in the ontologies, factors that can be considered as support in the generation of more natural phrases. Among these are semantic relationships between concepts that can help construct sentences in natural language. In this context, the current research aims to explore these properties for the generation of natural language for the English language from a set of templates developed by linguists and the use of lexical resources. Two evaluations were performed to evaluate criteria and variables for the proposed language generation algorithm and a third one for final validation of the research. The first evaluation sought to identify ways of generating natural language phrases from the RDF properties. Starting from the analysis of the results of the first evaluation, a new experiment was conducted to measure the naturalness of the sentences generated from the RDF properties.

Finally, a third evaluation was designed and executed, where linguistic professionals and native English speakers evaluated the short sentences generated by the algorithm. The results of the final evaluation were considered promising for applications that aim to generate natural language from the information of RDF properties with the support of lexical resources.

Keywords: Linked Open Data. Natural Language Generation. RDF.

LISTA DE FIGURAS

Figura 1 – Conjuntos de dados publicados no formato de dados abertos e ligados 27
Figura 2 – Arquitetura de um sistema GLN 31
Figura 3 – Exemplo de um subgrafo RDF de um relacionamento 50
Figura 4 – Sentença gerada a partir de uma tripla RDF e um *template* 54
Figura 5 – Fluxograma do algoritmo 57
Figura 6 – Componentes da implementação do algoritmo utilizando TAM 60
Figura 7 – Subgrafo contendo a tripla que gerou a sentença S3 94
Figura 8 – Subgrafo contendo a tripla que gerou a sentença S10 95
Figura 9 – Subgrafo contendo a tripla que gerou a sentença S19 98
Figura 10 – Esquema representativo do *template* utilizado para as sentenças 100
Figura 11 – Subgrafo contendo a tripla que gerou a sentença S20 104

LISTA DE GRÁFICOS

Gráfico 1 – Avaliação das sentenças geradas por C1 70
Gráfico 2 – Avaliação das sentenças geradas por C2 70
Gráfico 3 – Avaliação das sentenças geradas por C3 71
Gráfico 4 – Média das notas da avaliação das sentenças da segunda avaliação 74
Gráfico 5 – Avaliação da gramática do conjunto de Sentenças SAP 80
Gráfico 6 – Avaliação da gramática do conjunto de Sentenças SAL 80
Gráfico 7 – Avaliação da gramática do conjunto de Sentenças SIP 81
Gráfico 8 – Avaliação da gramática do conjunto de Sentenças SIL 81
Gráfico 9 – Avaliação da naturalidade do conjunto de Sentenças SAP 81
Gráfico 10 – Avaliação da naturalidade do conjunto de Sentenças SAL 82
Gráfico 11 – Avaliação da naturalidade do conjunto de Sentenças SIP 82
Gráfico 12 – Avaliação da naturalidade do conjunto de Sentenças SIL 82
Gráfico 13 – Avaliação da gramática do conjunto de Sentenças SAP 84
Gráfico 14 – Avaliação da gramática do conjunto de Sentenças SAL 84
Gráfico 15 – Avaliação da gramática do conjunto de Sentenças SIP 85
Gráfico 16 – Avaliação da gramática do conjunto de Sentenças SIL 85
Gráfico 17 – Avaliação da naturalidade do conjunto de Sentenças SAP 85
Gráfico 18 – Avaliação da naturalidade do conjunto de Sentenças SAL 86
Gráfico 19 – Avaliação da naturalidade do conjunto de Sentenças SIP 86
Gráfico 20 – Avaliação da naturalidade do conjunto de Sentenças SIL 86
Gráfico 21 – Avaliação da gramática do conjunto de Sentenças SAP 88
Gráfico 22 – Avaliação da gramática do conjunto de Sentenças SAL 88
Gráfico 23 – Avaliação da gramática do conjunto de Sentenças SIP 89
Gráfico 24 – Avaliação da gramática do conjunto de Sentenças SIL 89
Gráfico 25 – Avaliação da naturalidade do conjunto de Sentenças SAP 89
Gráfico 26 – Avaliação da naturalidade do conjunto de Sentenças SAL 90
Gráfico 27 – Avaliação da naturalidade do conjunto de Sentenças SIP 90
Gráfico 28 – Avaliação da naturalidade do conjunto de Sentenças SIL 90

LISTA DE QUADROS

Quadro 1 – Análise de trabalhos relacionados 43
Quadro 2 – Propriedades RDF 47
Quadro 3 – Marcações utilizadas nos *templates* 53
Quadro 4 – Exemplo de POST para o endpoint de questões 61
Quadro 5 – Consultas SPARQL utilizadas 62
Quadro 6 – Critérios para uma tripla com predicado válido para geração de linguagem 63
Quadro 7 – Conjunto de critérios para seleção de triplas RDF 68
Quadro 8 – Exemplos sentenças geradas na primeira avaliação 69
Quadro 9 – Exemplos sentenças geradas na segunda avaliação 72
Quadro 10 – Entidades utilizadas na segunda avaliação para geração de frases curtas 73
Quadro 11 – Relação de avaliadores 76
Quadro 12 – Conjunto de valores estatísticos para cada conjunto de respostas 79
Quadro 13 – As três sentenças com médias mais baixas para a gramática (SAP) 93
Quadro 14 – As três sentenças com médias mais baixas para a gramática (SAL) 97
Quadro 15 – As três sentenças com médias mais altas para a gramática (SAL) 97
Quadro 16 – As três sentenças com médias mais baixas para a gramática (SIP) 99
Quadro 17 – As três sentenças com médias mais baixas para a gramática (SIL) 101
Quadro 18 – As três sentenças com médias mais baixas para a naturalidade (SAP) 102
Quadro 19 – As três sentenças com médias mais baixas para a naturalidade (SAL) 103
Quadro 20 – As três sentenças com médias mais baixas para a naturalidade (SIP) 106
Quadro 21 – As três sentenças com médias mais baixas para a naturalidade (SIL) 107

LISTA DE TABELAS

Tabela 1 – Valores das métricas da avaliação da CR1 83
Tabela 2 – Valores das métricas da avaliação da CR2 87
Tabela 3 – Valores das métricas para Todos Participantes 91

LISTA DE SIGLAS

ACM	Association for Computing Machinery
API	Application Programming Interface
BDAC	Base de Dados Aberta e Conectada
BLEU	Bilingual evaluation understudy
C2T	Concept to text
CID	Classificação Estatística Internacional de Doenças e Problemas Relacionados com a Saúde
D2T	Data to text
DAC	Dados Abertos e Conectados
DDD	Discagem direta a distância
DDI	Discagem Direta Internacional
GLN	Geração de Linguagem Natural
GRU	Gated recurrent unit
HTTP	Hypertext Transfer Protocol
IEEE	Institute of Electrical and Electronics Engineers
LOD	Linked Open Data
MRR	Mean Reciprocal Rank
NLG	Natural Language Generation
OWL	Ontology Web Language
PLN	Processamento de Linguagem Natural
RDF	Resource Description Framework
ROUGE	Recall-Oriented Understudy for Gisting Evaluation
SPARQL	SPARQL Protocol and RDF Query Language
SW	Software
T2T	Text to text
TAM	Technical Achitecture Modeling
TER	Teoria da Estrutura Retória
URI	Uniform Resource Identifer
W3C	World Wide Web Consortium

SUMÁRIO

1 INTRODUÇÃO **20**
1.1 QUESTÃO DE PESQUISA 23
1.2 OBJETIVOS 24
1.2.1 Objetivo Geral **24**
1.2.2 Objetivos Específicos **24**
1.3 MÉTODO DE PESQUISA 24
1.4 ORGANIZAÇÃO DO TEXTO 25
2 FUNDAMENTAÇÃO TEÓRICA **26**
2.1 DADOS ABERTOS E CONECTADOS 26
2.2 GERAÇÃO DE LINGUAGEM NATURAL 29
2.3 ASPECTOS LINGUÍSTICOS 32
3 TRABALHOS RELACIONADOS **34**
3.1 MÉTODO DE PESQUISA BIBLIOGRÁFICA 34
3.1.1 Definição das questões de pesquisa **35**
3.1.2 Definição de filtros e critérios qualitativos **35**
3.1.3 Definição da *String* de busca **36**
3.2 DESCRIÇÃO DOS TRABALHOS 36
3.2.1 Sistemas baseados em *templates* **36**
3.2.2 Sistemas baseados em aprendizagem de máquina **38**
3.2.3 Sistemas híbridos **39**
3.3 Comparativo **41**
4 ALGORITMO **46**
4.1 INFORMAÇÕES CONTIDAS NA TRIPLA RDF 46
4.1.1 Uso da propriedade *rdfs:label* **48**
4.1.2 Uso da propriedade *rdfs:range* **48**
4.1.3 Uso da propriedade *rdf:type* **49**
4.1.4 Uso da propriedade *rdfs:comment* **50**
4.1.5 Uso da propriedade *rdfs:subPropertyOf* **51**
4.1.6 Exemplo prático do uso das propriedades RDF **51**
4.2 *TEMPLATES* 51
4.3 FLEXIBILIZAÇÃO E DIVERSIFICAÇÃO NA GERAÇÃO DE SENTENÇAS 54
4.4 DESCRIÇÃO DO ALGORITMO E SUAS ETAPAS 55

4.4.1 Exemplo ilustrativo **58**
5 IMPLEMENTAÇÃO **59**
5.1 ESTRUTURA 59
5.2 FUNCIONAMENTO 61
5.2.1 Substituição das marcações de um *template* linguístico **63**
6 AVALIAÇÃO **66**
6.1 PRIMEIRA AVALIAÇÃO: VALIDAÇÃO DOS CRITÉRIOS PARA SELEÇÃO DE TRIPLAS PARA GERAÇÃO DE FRASES CURTAS 66
6.2 SEGUNDA AVALIAÇÃO: INCLUSÃO DE RECURSOS LINGUÍSTICOS 72
6.3 TERCEIRA AVALIAÇÃO: ANÁLISE FINAL 75
6.3.1 Avaliadores **75**
6.3.2 Modelo de avaliação **77**
6.3.3 Método de análise **77**
6.3.4 Apresentação dos resultados **79**
6.3.4.1 Falantes nativos (CR1) 79
6.3.4.2 Falantes não-nativos (CR2) 84
6.3.4.3 Todos os participantes (CR3) 88
6.3.5 Análise dos resultados **91**
6.3.5.1 Gramática do conjunto de sentenças afirmativas para entidades da classe pessoa (SAP) 92
6.3.5.2 Gramática do conjunto de sentenças afirmativas para entidades da classe lugar (SAL) 96
6.3.5.3 Gramática do conjunto de sentenças interrogativas para entidades da classe pessoa (SIP) 99
6.3.5.4 Gramática do conjunto de sentenças interrogativas para entidades da classe lugar (SIL) 101
6.3.5.5 Naturalidade do conjunto de sentenças afirmativas para entidades da classe pessoa (SAP) 101
6.3.5.6 Naturalidade do conjunto de sentenças afirmativas para entidades da classe lugar (SAL) 103
6.3.5.7 Naturalidade do conjunto de sentenças interrogativas para entidades da classe pessoa (SIP) 105
6.3.5.8 Naturalidade do conjunto de sentenças interrogativas para entidades da classe pessoa (SIL) 107

6.3.6 Discussão dos resultados ..... 108~~108~~
7 CONSIDERAÇÕES FINAIS ..... 111~~111~~
7.1 CONTRIBUIÇÕES ..... 112~~112~~
7.2 TRABALHOS FUTUROS ..... 113~~113~~
REFERÊNCIAS ..... 115~~115~~
APÊNDICE A – TEMPLATES UTILIZADAS PELO ALGORITMO ..... 120~~120~~
APÊNDICE B – CONJUNTO DE SENTENÇAS AVALIADAS NA PRIMEIRA AVALIAÇÃO ..... 121~~121~~
APÊNDICE C - CONJUNTO DE SENTENÇAS AFIRMATIVAS GERADAS NA SEGUNDA AVALIAÇÃO ..... 126~~126~~
APÊNDICE D – CONJUNTO DE SENTENÇAS INTERROGATIVAS GERADAS NA SEGUNDA AVALIAÇÃO ..... 128~~128~~
APÊNDICE E – CONJUNTO DE SENTENÇAS AFIRMATIVAS DA TERCEIRA AVALIAÇÃO ..... 131~~131~~
APÊNDICE F – CONJUNTO DE SENTENÇAS INTERROGATIVAS DA TERCEIRA AVALIAÇÃO ..... 133~~133~~

1 INTRODUÇÃO

Desde a criação do conceito de Dados Abertos e Conectados uma série de iniciativas vêm ocorrendo no sentido de estimular que instituições, tanto privadas, quanto governamentais, disponibilizem seus dados para acesso global através da Internet (HEATH e BIZER, 2011; ISOTANI e BITTENCOURT, 2015). Segundo dados do site *The Linked Open Data Cloud* (MCCRAE, 2018), que mantém uma atividade periódica de atualização sobre este assunto, atualmente há mais de mil bases de dados abertos e conectados. Essas iniciativas, na maioria das vezes, buscam tornar públicas as informações das instituições em função do fomento à transparência de suas ações.

Além das iniciativas para publicação de dados em formato aberto, também se observa pesquisas explorando o assunto com diferentes objetivos, tais como Duma e Klein (2013), que buscaram descrever fatos sobre entidades descritas na Wikipedia, e Kaffee et al. (2018), que buscaram gerar sumários a partir de dados da WikiData. Outras pesquisas fomentam a criação de algoritmos que representam o conhecimento disponibilizado a partir de aplicações WEB em grafos RDF (*Resource Description Framework*[1]). Um dos exemplos mais conhecidos dessa iniciativa é a DBPedia (LEHMANN et al., 2015), projeto que converte os trechos de artigos textuais da Wikipedia para equivalentes em formatos de dados estruturados com o padrão RDF.

Atualmente, o conjunto de dados disponibilizados em formato anotado, ou seja, o conjunto das bases de dados abertos e conectados, representa grande volume de informações. Como exemplo, a análise de algumas destas bases ilustra que já podem ser acessadas quantidades entre 10 milhões de conceitos (YAGO3[2]), 16 milhões de conceitos (DBpedia[3]) ou até 20 milhões de conceitos (Wikidata[4]). Como consequência desta disponibilidade, observa-se a possibilidade de utilização de bases de dados abertos e conectados em larga escala, como fonte de informações para sistemas de pergunta e resposta (SHEKARPOUR et al., 2016) ou como apoio para a geração de linguagem natural (INDURKHYA e DAMERAU, 2010; PERERA e NAND, 2017).

[1] O RDF é um framework para modelagem de dados que descreve recursos através de tuplas contendo relações com as características destes recursos (W3C, 2014b).
[2] http://yago-knowledge.org/
[3] http://www.dbpedia.pt/
[4] https://www.wikidata.org/wiki/Wikidata:Main_Page

Historicamente, boa parte dos mecanismos para geração de linguagem natural utiliza grandes conjuntos de documentos, analisados por algoritmos de aprendizagem de máquina para a geração de modelos capazes de apoiar na criação de sentenças de texto em linguagem natural (INDURKHYA; DAMERAU, 2010). Por exemplo, supondo que haja uma pergunta em linguagem natural para buscar a idade de uma personalidade famosa, sistemas com esta abordagem extraem as principais palavras-chave da pergunta e em seguida buscam por essas palavras em um grande corpus de documentos aplicando algoritmos com o objetivo de gerar uma frase com a resposta. Esta abordagem, apesar de proporcionar bons resultados, pode gerar dependências indesejáveis quanto à disponibilidade das bases de dados a serem utilizadas como fontes para este processo.

Segundo Reiter e Dale (1997), dentre os diversos subsídios para a geração automática de linguagem natural encontram-se as informações para planejamento do texto a ser construído, o que pode ser fundamental para a geração correta e também para a geração de acordo com o uso da linguagem. Em geral os métodos baseados unicamente em corpus apresentam dificuldades para atender a estes subsídios, sendo que, ao contrário, as bases de dados abertos e conectados disponibilizam um conjunto de conceitos com relações e semântica bem definidos, o que favorece o funcionamento dos algoritmos de geração de linguagem natural (HÖFFNER et al. 2017).

A partir deste contexto, observa-se a existência de possibilidades para explorar bases de dados conectados para geração de linguagem natural. Atualmente, nota-se na literatura, propostas em três categorias, ou seja, aquelas que utilizam *templates*, as que empregam técnicas de aprendizado de máquina e as que se caracterizam como arquiteturas híbridas (GATT; KRAHMER, 2018). A primeira abordagem utiliza regras expressas em modelos de estruturas de frases (*templates*) que representam futuras sentenças para geração de linguagem natural. Ela foi utilizada nos trabalhos de Moussallem et al. (2018) para geração de textos em português. Ela também foi citada por Perera, Nand e Naeem (2017) como exemplo para geração de respostas a questionamentos apresentados em sistemas de perguntas e respostas. Esta abordagem, no entanto, possui uma tendência a gerar um conjunto de frases semelhantes, uma vez que os *templates* devem ser reutilizados. A segunda abordagem utiliza de técnicas de aprendizado de máquina, empregando redes neurais e outras abordagens estatísticas, para composição de modelos a partir de grande

conjunto de documentos para geração de linguagem natural (VOUGIOUKLIS, 2017). Esta abordagem, entretanto, necessita de um grande conjunto de documentos e grande poder de processamento para treinamento de redes neurais. Por fim, a terceira abordagem tira proveito de aspectos positivos existentes em ambas as técnicas (PERERA; NAND, 2015). Normalmente, é encontrada em sistemas sequenciais, onde uma etapa pode ou não complementar a outra.

Desta forma, a existência de volumes cada vez maiores de dados em formato anotado semanticamente com o padrão RDF, em especial nas Bases de Dados Abertos e Conectados, permite que os sistemas de geração de linguagem natural utilizem estes recursos para a geração de sentenças em linguagem natural. O principal ponto favorável neste sentido é a existência de relações bastante definidas entre os conceitos e os demais dados sobre estes conceitos, o que proporciona a estrutura necessária para os algoritmos atuais (RODRIGUES, 2017). O formato RDF possibilita que um grande número de propriedades possa ser definido e relacionado com outras propriedades. Desta forma, o processo de caminhamento sobre um grafo RDF permite que o conhecimento sobre um determinado conceito possa ser descoberto e relacionado de forma direta com todos os aspectos associados a este conceito.

Observa-se, entretanto, que as abordagens estudadas exploram minimamente esse potencial. O objetivo deste trabalho é atuar nesta lacuna, verificando se o emprego de informações das estruturas de dados abertos e conectados para gerar frases curtas em linguagem natural a partir de *templates* linguísticos melhora a naturalidade das sentenças.

Além disso, este trabalho difere dos demais trabalhos estudados em pelo menos em quatro aspectos. Primeiramente, ao não se limitar a um número reduzido de propriedades das triplas RDF, tais como o uso apenas das propriedades *type* e *label*. O algoritmo proposto, além das propriedades já mencionadas, faz uso das demais propriedades listadas na definição do RDF (RDF, 2014a), tais como *range, domain, comment*, entre outras. Esta característica possibilita ao algoritmo o uso de um maior número de opções para o planejamento da geração das frases, diversificando os resultados.

Em um segundo aspecto, o algoritmo utiliza um conjunto de *templates* construído com apoio de profissionais da linguística, buscando incorporar desta forma um conhecimento aprofundado das questões de uso da linguagem no cotidiano. Essa

abordagem foi adotada como uma hipótese para o apoio na geração de sentenças com alto grau de naturalidade, uma vez que o corpus de *templates* foi desenvolvido por profissionais que têm um maior domínio da linguagem e suas teorias. Ao optar por *templates* e não pelo aprendizado de máquina, por exemplo, inibe-se a tendência de linguagem que um algoritmo baseado em uma coleção de documentos possa vir a desenvolver. Por exemplo, dependendo da coleção de documentos que uma técnica de aprendizado de máquina utiliza para operações de treinamento, podem ser observados impactos relacionados com o conteúdo destes documentos, como o emprego de palavras não-usuais para expressão de determinados conceitos.

Como terceiro aspecto, o algoritmo busca gerar maior diversidade e naturalidade de sentenças para um determinado conjunto de informações através do uso de léxicos de sinônimos e léxicos de combinações de palavras. Estes conjuntos de recursos permitem que a etapa final de lexicalização seja definida de forma a gerar resultados diversificados, que utilizem um maior repertório linguístico. Os *templates* foram criados para o idioma Inglês americano e o algoritmo foi configurado para se conectar ao capítulo em Inglês da DBPedia, pois o conjunto de triplas e informações é o mais completo dentre os capítulos disponíveis.

Por fim, como quarto aspecto a destacar, um dos métodos utilizados para a análise dos resultados na avaliação final é diferente dos apresentados por trabalhos relacionados ao mesmo tema. Diferentemente dos demais, além de avaliar com especialistas as frases curtas geradas pelo algoritmo, a concordância dos avaliadores em relação à qualidade das sentenças também foi avaliada. Desta forma, foi possível identificar a intensidade de concordância dos avaliadores em relação às notas atribuídas às sentenças.

1.1 QUESTÃO DE PESQUISA

Como mencionado anteriormente, durante o estudo realizado, observou-se uma lacuna na utilização de bases de dados abertos e conectados para a geração de frases curtas em linguagem natural. Apesar destas bases disponibilizarem um conjunto rico de relacionamentos entre os conceitos mantidos, estes relacionamentos são pouco empregados ainda em sistemas de geração de linguagem natural.

Desta forma, esse trabalho visa responder a seguinte questão de pesquisa: O uso das informações presentes nas estruturas de bases de dados abertos e conectados apoiadas por *templates* e léxicos linguísticos pode gerar sentenças curtas em linguagem natural com um grau adequado de naturalidade?

1.2 OBJETIVOS

Nesta seção serão descritos os objetivos geral e específicos desta pesquisa.

1.2.1 Objetivo Geral

O objetivo geral deste trabalho é verificar se o emprego de informações das estruturas de dados abertos e conectados para gerar frases curtas em linguagem natural a partir de *templates* linguísticos melhora a naturalidade das sentenças.

1.2.2 Objetivos Específicos

Para operacionalizar e atingir o objetivo geral mencionado anteriormente, quatro objetivos específicos foram definidos. Os quatro objetivos específicos listados abaixo podem ser considerados subprodutos desta pesquisa:

a) Composição de um conjunto de *templates* linguísticos;
b) Descritivo do potencial de uso das propriedades RDF para uso em sistemas de geração de linguagem natural a partir de dados abertos e conectados;
c) Desenvolvimento da arquitetura e implementação de um algoritmo para geração de linguagem natural a partir de dados abertos e conectados apoiados por recursos linguísticos explorando propriedades RDF;
d) Avaliação do material gerado pelo algoritmo proposto utilizando linguistas.

1.3 MÉTODO DE PESQUISA

Esta pesquisa visa implementar melhorias na geração de sentenças próximas à linguagem natural do ser humano a partir de bases de dados abertos e conectados utilizando *templates* linguísticos. Quanto aos procedimentos técnicos, esta pesquisa se caracteriza como uma pesquisa experimental, uma vez que o projeto do algoritmo

de geração de linguagem terá variáveis de seu fluxo alteradas buscando aumentar a percepção de naturalidade das sentenças geradas. Quanto aos objetivos, o trabalho se caracteriza como uma pesquisa explicativa. Para tanto, foi adotado Wazlawick (2017) para definir as seguintes etapas como método de trabalho: estudos e análise de trabalhos relacionados; projeto de solução com as informações coletadas nas etapas anteriores; descrição do protótipo e uma abordagem para avaliação.

Dentro deste contexto, um algoritmo que produza linguagem natural, a partir de *templates* linguísticos, explorando bases de dados abertos e conectados foi desenvolvido e implementado e, três avaliações foram realizados. Preliminarmente, duas avaliações validaram o resultado do processo de geração de sentenças afirmativas e interrogativas em inglês tendo como objeto alguns conceitos dentro da DBPedia. A primeira avaliação validou a forma na qual a seleção dos relacionamentos de uma tripla poderia gerar informações precisas e corretas em linguagem natural. A segunda avaliação buscou avaliar a naturalidade das sentenças geradas. Já a terceira avaliação buscou a geração de frases curtas natural e gramaticalmente corretas a partir dos resultados encontrados nas duas primeiras avaliações. A terceira avaliação também foi avaliada por linguistas americanos e brasileiros com o objetivo de validar a qualidade das sentenças geradas. Além de avaliar a nota dada para cada sentença, também foi mensurada a concordância entre os avaliadores para com os objetos avaliados.

O algoritmo produzido até então tem apoio de um grupo de linguistas que gerou um conjunto de *templates* linguísticos para uso do algoritmo para criar sentenças afirmativas e interrogativas em linguagem natural.

1.4 ORGANIZAÇÃO DO TEXTO

Nos próximos capítulos, serão apresentados os conceitos principais do tema deste trabalho. Em seguida será descrito como outros pesquisadores estão trabalhando com o mesmo tema e onde essa pesquisa se diferencia. Após, o algoritmo, sua implementação e as avaliações serão descritas e discutidas. Por fim, uma conclusão sobre a pesquisa e eventuais trabalhos futuros serão apresentados.

2 FUNDAMENTAÇÃO TEÓRICA

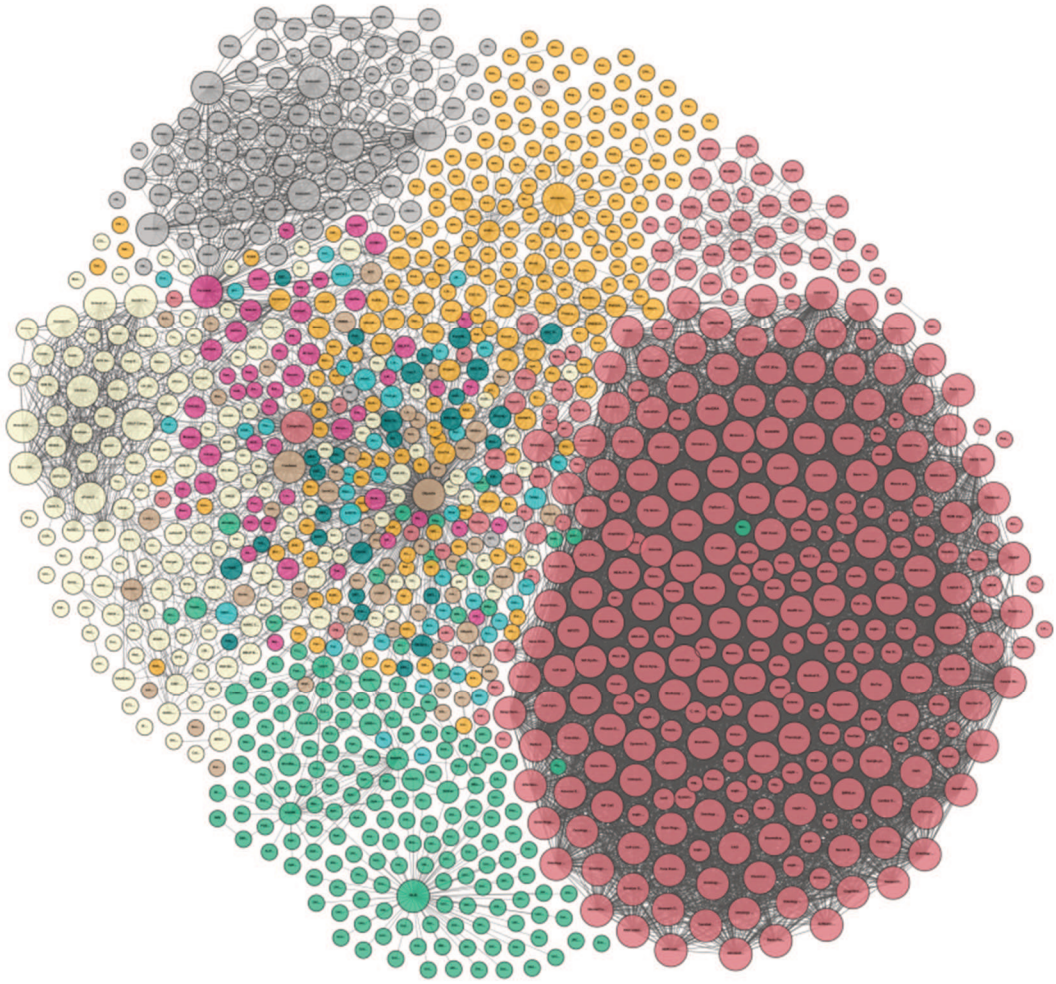
A construção e elaboração do algoritmo necessitou do suporte de diversos aspectos das áreas da ciência da computação e linguística. Desta forma, essa seção apresenta esses conceitos que serão abordados no decorrer desta dissertação. Inicialmente será apresentada a descrição de Dados Abertos e Conectados. Após, será descrito o processo de geração de linguagem natural e, por fim, os principais aspectos linguísticos abordados por este trabalho também serão descritos.

2.1 DADOS ABERTOS E CONECTADOS

Todos os dias, novos conteúdos são produzidos e publicados na Internet. Porém, esses conteúdos não são estruturados para sistemas computacionais. Seres humanos conseguem ler o material e compreender os tópicos tratados. No entanto, computadores têm dificuldade, pois a informação não está estruturada de um modo que seja passível de compreensão por sistemas computacionais (PERERA et al., 2017). Com o intuito de estruturar e relacionar essas informações para que possam ser consultadas por aplicações, surgiu a Web Semântica. O conceito de Web Semântica, também chamado de Web dos Dados, surgiu da necessidade de possibilitar que computadores façam maior proveito dos dados disponíveis na Internet (WEB, 2011). A Web Semântica é suportada por uma série de tecnologias, dentre elas está o conceito de Dados Abertos e Conectados.

Dados Abertos e Conectados (ou ligados), também conhecido em inglês como *Linked Open Data*, são um conjunto gigantesco de dados e informações interconectadas. Esses dados devem estar disponíveis em formato padronizado, acessível e gerenciável por ferramentas semânticas. (LINKED, 2015). Além disso, os dados disponíveis podem conter relacionamentos entre eles e entre dados de outras bases. Desta forma, BDAC podem estar conectadas com outras BDAC formando uma nuvem de dados conectados (Figura 1) (UNGER et al., 2014).

Figura 1 – Conjuntos de dados publicados no formato de dados abertos e ligados.



Fonte: MCCRAE, 2018.

Os princípios do conceito de Dados Abertos e Ligados foram descritos por Tim Berners-Lee em 2006. São quatro regras (LINKED, 2006):

1. Uso de URI para identificar entidades/coisas.
2. Uso de HTTP URIs para localizar as entidades/coisas.
3. Prover informações úteis utilizando padrões como RDF e SPARQL quando uma entidade/coisa for localizada.
4. Incluir *links* para outras URIs para que mais entidades/coisas possam ser descobertas.

Dentre as tecnologias necessárias para manutenção dos princípios estão OWL, RDF, SPARQL. OWL é uma linguagem para definição de ontologias na WEB.

Seguindo a descrição de OWL definida pela W3C, OWL é uma linguagem com o objetivo de possibilitar descrever classes e relações entre entidades/coisas (OWL, 2004). O papel da OWL dentro dos conjuntos que fundamentam o conceito de Dados Abertos e Conectados é fornecer um vocabulário que ajuda na integração dos dados e na inferência de novos relacionamentos (ONTOLOGIES, 2015).

RDF é um *framework* que auxilia na descrição das informações a serem representadas dentro de um conjunto de dados. RDF também descreve todas as relações entre as informações (RDF, 2014a). Mais especificamente, RDF é um modelo de dados que descreve informações como nodos e arcos dirigidos. Uma informação pode ser representada por uma ou mais triplas (*triples)*. Uma tripla é composta por três partes: sujeito, predicado e objeto. Desta forma, um conjunto de informações em triplas compõem um grafo RDF (HEATH e BIZER, 2011).

A especificação da modelagem de dados RDF descreve o conceito de propriedades RDF como a relação entre um sujeito e um objeto (RDF, 2014a). Dentro da especificação também são descritas propriedades padrões que podem ser utilizadas em uma modelagem de dados. Elas são instâncias da classe *rdf:property*. Na descrição do RDF há 7 propriedades (RDF, 2014a):

- *rdfs:range*: Essa propriedade é usada para afirmar que o valor de uma propriedade é instância de uma ou mais classes. Por exemplo, a tripla *P → rdfs:range → C* afirma que P é uma instância da classe *rdf:property* e C é uma classe RDF. Isso implica que quaisquer que seja o objeto denotado por P, ele é instância da classe C.
- *rdfs:domain*: Essa propriedade é usada para afirmar que um recurso que tem uma propriedade é instância de uma ou mais classes. *P → rdfs:domain → C* afirma que P é uma propriedade e C uma classe e que todos as instâncias com predicado P são instâncias de C. Em outras palavras, *rdfs:domain* é o inverso da propriedade *rdfs:range*.
- *rdf:type*: Essa propriedade é utilizada para afirmar que um recurso é uma instância de uma ou mais classes.
- *rdfs:subClassOf*: Essa propriedade é usada para afirmar que todas as instâncias de uma classe são instâncias de outra. A tripla *C1 → rdfs:subClassOf → C2* afirma que C1 e C2 são classes e que C1 é uma subclasse de C2.

- *rfds:subPropertyOf*: Essa propriedade é usada para afirmar que todos os recursos relacionados por uma propriedade são também relacionados por outra. A tripla *P1 → rdfs:subPropertyOf → P2* afirma que P1 e P2 são propriedades e que P1 é uma sub-propriedade de P2.
- *rdfs:label*: Essa propriedade é usada para fornecer uma versão legível do nome do recurso para seres humanos.
- *rdfs:comment*: Essa propriedade é usada para fornecer uma descrição legível do recurso para seres humanos.

Sendo fundamental na definição de Dados Abertos e Conectados, URI (*Uniform Resource Identifer)* é um nome usado para identificar e/ou localizar um recurso na Internet. Todos os componentes de uma tripla podem ser um URI. No entanto, o objeto da tripla pode ser um valor literal como um número, data, texto, etc. Portanto, existem basicamente dois tipos de triplas RDF: Triplas literais e *Links* RDF. A primeira servindo para descrever nomes e dados estáticos (data de nascimento, sobrenome, etc.). Além disso, também podem conter *tags* para identificar o idioma natural ou podem conter um URI que identifica o tipo de dado (ponto flutuante, número inteiro, etc.). A segunda, por outro lado, descreve uma relação entre dois recursos. Também é possível categorizar *Links* RDF como internos e externos. Os *Links* RDF internos apontam para recursos dentro da mesma fonte do conjunto de dados. *Links* RDF externos apontam para recursos em outras fontes de dados (HEATH e BIZER, 2011).

No entanto, essas bases têm de ser passíveis de consulta. Para consultar base de dados abertos e conectados modeladas com RDF, foi padronizada uma linguagem de consulta chamada SPARQL pela W3C. SPARQL possibilita a consulta de informações através de um padrão de triplas onde o sujeito, predicado e objeto da consulta podem ser variáveis (SPARQL, 2008).

2.2 GERAÇÃO DE LINGUAGEM NATURAL

O processamento de linguagem natural tem como objetivo a interação entre máquinas e seres humanos. O processamento de linguagem natural envolve disciplinas de linguística, inteligência artificial e ciência da computação (GUPTA, 2014). Dentro da área de pesquisa de processamento de linguagem natural, há uma área de estudo para geração de linguagem natural. Essa área tem como objetivo

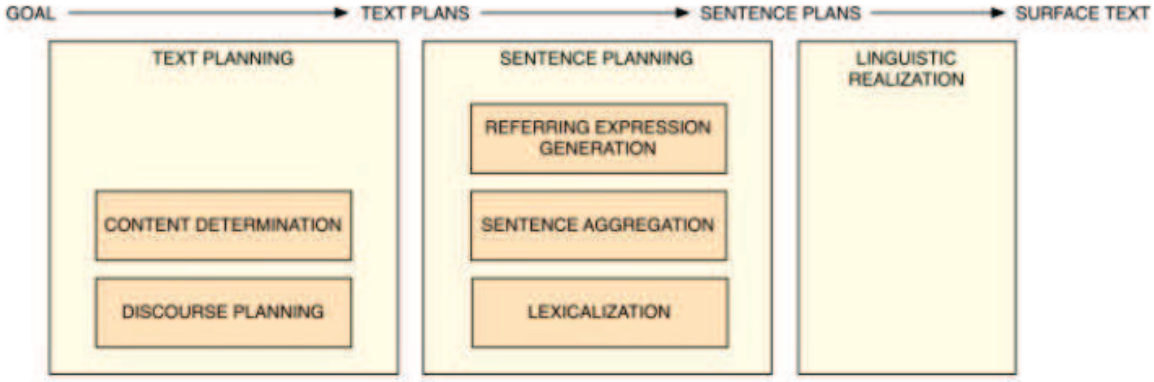
produzir textos em linguagem natural com ótimo nível de precisão e entendimento a partir de informações armazenadas em sistemas computacionais (PERERA e NAND, 2017).

Antes da década de 1980, os sistemas relacionados ao processamento de linguagem eram desenvolvidos a partir de regras escritas à mão. A partir da década de 1980, a descoberta de técnicas de aprendizado de máquina para processamento de linguagem natural iniciou uma revolução (GUPTA, 2014). Os primeiros sistemas de geração de linguagem traduziam dados em simples textos com baixíssima ou pouca variação. No entanto, atualmente sistemas de geração de linguagem natural ganharam mais complexidade e envolvem tópicos de linguística e metodologias diferenciadas para geração de texto.

De forma geral, sistemas de geração de linguagem natural podem ser diferenciados por dois critérios: a entrada dos sistemas e o seu objetivo de comunicação (VICENTE et al., 2015.). A entrada do sistema pode ser do tipo D2T e T2T. O primeiro tipo é a conversão de dados em geral (dados numéricos, dados estruturados, etc.) para texto. Já o segundo tipo é a conversão de texto para texto, sendo a entrada do sistema textos ou um conjunto de sentenças que seriam manipulados a fim de gerar um novo texto. Avaliando o objetivo de comunicação, sistemas de geração de linguagem natural podem ser classificados como geradores de textos informativos, sumários, simplificadores textuais, geradores de textos persuasivos, sistemas de diálogo, sistemas explanatórios e sistemas de recomendação (VICENTE et al., 2015.). No entanto, independente da classificação, um sistema gerador de linguagem segue um fluxo de tarefas para atingir seu objetivo de comunicação.

Um sistema Gerador de Linguagem Natural (GLN) consiste de uma série de tarefas responsáveis por fazer o tratamento dos seus dados de entrada e gerar o texto final que será apresentado aos seus utilizados (AIRES, 2016). Uma maneira abstrata de entender a arquitetura de um sistema gerador de linguagem natural é separar seus processos em três fases: planejador de textos (*Text Planning*), planejador de sentenças (*Sentence Planning*) e síntese de discurso (*Speech Synthesis*). Cole et al., em 1997, chamaram essa etapa de *Speech Synthesis*, enquanto Reiter e Dale, também em 1997, chamaram de *Linguistic Realization* (Figura 2).

Figura 2 – Arquitetura de um sistema GLN.



Fonte: REITER e DALE, 1997.

A fase de planejamento de texto é responsável por organizar, determinar e estruturar as informações que irão aparecer no texto final produzido pelo sistema. As tarefas de organização e determinação (*Content Determination*) do conteúdo selecionam as informações que irão para a próxima etapa do gerador de linguagem natural. Já a tarefa de estruturação (*Discouse Planning*) estabelece a ordem das informações com o objetivo de deixar o texto final coerente (AIRES, 2016).

A fase de planejamento de sentenças é responsável por estabelecer as informações que irão aparecer nas sentenças e por quais palavras elas serão expressadas. Para tanto, há três grupos de passos distintos. O primeiro agrupa as informações em uma sentença escolhendo a forma sintática que será usada para combiná-las. Após, a fase de lexicalização é responsável por escolher a palavra correta para representar as informações da futura sentença. Por fim, as sentenças passam por uma última tarefa nesta fase para eliminar ambiguidade do texto. Essa tarefa determina a melhor forma de referenciar conceitos e objetos que podem repetir dentro do texto (AIRES, 2016).

Por fim, a tarefa de síntese do discurso (*Speech Synthesis*) aplica as regras gramaticais para produzir o texto de forma que ele esteja sintaticamente, morfologicamente e ortograficamente correto. (RAMOS-SOTO, 2016).

Para cada uma dessas etapas, existem diferentes abordagens de implementação. De modo geral as abordagens podem ser divididas em três grupos: métodos baseados em conhecimento, métodos estatísticos e métodos híbridos. O primeiro usa recursos para geração de seu texto. Esses recursos têm forte influência

linguística, tais como uso de dicionários, enciclopédias, conjuntos de regras linguísticas ou *templates* pré-definidas. Já o segundo tem como sua fonte um corpus de texto anotado ou não, e as probabilidades associadas a cada sentença. Normalmente nessa abordagem são utilizadas técnicas de aprendizagem de máquina para mineração de *templates* para geração de sentenças. A abordagem híbrida, por sua vez, combina métodos das duas abordagens anteriores (AIRES, 2016).

2.3 ASPECTOS LINGUÍSTICOS

Conforme já mencionado anteriormente, a geração de linguagem natural é uma área que, além de tópicos de inteligência artificial e ciência da computação, também aborda alguns aspectos linguísticos. De acordo com Santos (1994), a "linguística é o estudo científico da linguagem e das línguas naturais e seus discursos". Desta forma, a área da linguística contém modelos teóricos e recursos linguísticos para construções textuais (SANTOS, 1994). Dentre esses, os recursos da linguística computacional são um conjunto de ferramentas passíveis de utilização por sistemas geradores de linguagem natural, tais como léxicos e *parsers*.

Léxicos são recursos primários para a engenharia da linguagem. As informações contidas em um léxico podem conter vários níveis linguísticos, dentre eles estão: morfologia, sintaxe e semântica. A partir dessas informações, um sistema gerador de linguagem natural é capaz de fazer o correto tratamento de palavras para expressar uma informação (ZAVAGLIA, 2003). Já *parsers* são *softwares* que fazem o tratamento sintático das línguas naturais. Eles atribuem uma estrutura e interpretação para uma sequência linguística (XAVIER e MATEUS, 1992).

Existem diversos léxicos computacionais disponíveis para construção de sistemas GLN. Dentre eles um dos mais famosos é o WordNet. WordNet é uma base léxica para o idioma inglês mantido pelo grupo de pesquisa em processamento em linguagem natural da Universidade de Princeton. O WordNet contém verbos, substantivos, adjetivos em inglês organizados em conjuntos de sinônimos, cada uma representando um conceito lexicalizado (MILLER, 1995). Além disso, os conjuntos de sinônimos (*synsets*) são ligados por relações semânticas. No entanto, ainda existem léxicos especializados em outros aspectos linguísticos.

Existem léxicos especializados em colocações verbais. Colocações verbais são um fenômeno léxico que cobrem palavras e frases que são comumente utilizadas na língua sem uma regra geral sintática ou semântica aplicada (MCKEOWN e RADEV, 2000). O dicionário Oxford, por exemplo, tem um léxico de colocações. Esse dicionário pode ser adquirido ou acessado de forma gratuita na Internet (OXFORD, 2018).

Já na definição da naturalidade, existe um dilema notório, uma vez que é uma noção inerente às nossas vidas. De acordo com Weisser (1997), muitos estudiosos usam esse termo, bem como "fluência" e "proficiência oral", sem ter uma compreensão clara do que essas noções significam. Embora alguns dicionários tragam suas definições, os conceitos supracitados podem ser entendidos como abstratos.

A discussão sobre o conceito de naturalidade não é uma novidade. É datado de Platão e pode ser visto no campo da linguística a partir da perspectiva da fonologia. Por exemplo, no final dos anos 70, Stampe (1979) e Donegan e Stampe (1979) desenvolveram uma teoria chamada “Fonologia Natural”. Segundo esta teoria, "as condições do uso da linguagem (performance) são responsáveis pela natureza da linguagem" (STAMPE, 1979, p. 43). Isso significa que a cultura e variação linguística pode afetar a fala do orador e da maneira que ele / ela cria uma frase. Além disso, Stampe (1979, p. 35) afirma que os fonemas mentais "são representações mentais de sons que são, pelo menos em princípio, pronunciáveis".

Dadas essas definições, Weisser (1997) apresenta alguns critérios para definir o que é naturalidade. Essas são:

i. Adequação - redação eloquente e uso de expressões;
ii. Gramaticalidade - uso coerente de regras gramaticais, mesmo que nem sempre seja produzido sentenças gramaticais;
iii. Autenticidade - fala espontânea.

Nesse sentido, o autor afirma que, apesar de ter uma frase perfeitamente escrita em termos de gramática, isso não indica que essa frase é natural. O autor também afirma que um corpus pode conter sentenças não-naturais e artificiais, mesmo que seja composto de dados genuínos atestados. Isso ocorre porque, às vezes, na linguagem escrita, o foco é centralizado na formalidade que é esquecido a espontaneidade das conversas diárias.

3 TRABALHOS RELACIONADOS

Os trabalhos relacionados ao tema desta pesquisa foram selecionados de duas formas. Primeiramente, ocorreu a leitura de pesquisas na área para embasamento sobre o tema e o estado da arte das tecnologias de bases RDF e geração de linguagem natural. Com esse conjunto inicial de artigos, buscou-se explorar a área e identificar possíveis lacunas e tendências sobre os dois temas. A partir dessa leitura, validou-se o modo como a linguagem natural estava sendo gerada a partir de dados abertos e conectados. Porém, uma pesquisa bibliográfica mais detalhada e sistematizada, com artigos mais recentes, foi realizada com o intuito de identificar o cenário dos trabalhos atuais sobre o tema.

Desta forma, este capítulo está dividido em três subseções. A primeira subseção descreve o método utilizado para pesquisa bibliográfica detalhada de artigos recentes sobre o tema; a segunda apresenta um resumo dos trabalhos relacionados estudados e já organizados em categorias de acordo com suas características; por fim, a terceira apresenta uma análise comparativa dos trabalhos mais significativos estudados.

3.1 MÉTODO DE PESQUISA BIBLIOGRÁFICA

O método é uma simplificação do método descrito em uma revisão sistemática de literatura (KITCHENHAM e CHARTERS, 2007). Em suma, o método compreende os seguintes passos:

1) Definição das questões de pesquisa que guiarão o estudo;
2) Especificação das palavras-chaves e fontes de consulta que serão utilizadas para obter material para o estudo;
3) Definição dos filtros que serão usados para reduzir o material obtido a partir do passo 2;
4) Definição dos critérios qualitativos para elencar ou eliminar o material obtido a fim de obter somente os trabalhos de pesquisa relevantes.

Seguindo esta lógica, as próximas subseções detalham cada um dos passos previstos neste método.

3.1.1 Definição das questões de pesquisa

O objetivo deste estudo é obter informações sobre o estado da arte de algoritmos e modelos que geram linguagem natural a partir de bases de dados abertos e conectados. Com base neste objetivo principal, foram definidas as seguintes questões:

- Quais são os tipos de arquiteturas disponíveis para sistemas de GLN a partir de DAC?
- Quais são as possíveis aplicações para sistemas GLN a partir de DAC?
- Quais são as técnicas e métodos que são utilizados para gerar linguagem a partir de DAC?

3.1.2 Definição de filtros e critérios qualitativos

Essas questões sem nenhum filtro retornaram uma quantidade significativa de trabalhos. No entanto, buscava-se obter o estado da arte do tema objeto deste estudo. Para isso, o primeiro filtro aplicado foi a limitação do ano de publicação para 2013. Logo, somente artigos publicados a partir de 2013 até 2018 seriam objeto de estudo. Desta forma, o número de artigos foi reduzido, de acordo com as bases de dados consultadas, para: ACM (3), Springer (53) e Google Scholar (488).

O próximo passo foi avaliar os artigos de acordo com seus resumos. Para essa avaliação, foram levadas em consideração o uso de bases de dados abertos e conectados para geração de linguagem natural e que continham métodos para avaliação dos resultados e/ou dos possíveis resultados a partir do método aplicado. Destes artigos analisados, um total de nove artigos foram selecionados. Esses trabalhos foram classificados de acordo com o uso de base de dados abertos e conectados para geração de linguagem através do apoio de redes neurais, *templates* e através de métodos híbridos. O número de citações não teve peso substancial para seleção dos trabalhos relacionados, pois trabalhos com esse tema são recentes na literatura.

3.1.3 Definição da *String* de busca

Para fonte de consulta de trabalhos de pesquisa, foram escolhidos os repositórios *online* da ACM, IEEE, Springer e o mecanismo de busca do Google Scholar. A *string* de busca utilizada foi**: *LOD OR "Linked Data" OR "Linked Open Data") AND ("Natural Language Generation" OR NLG)***, que busca especificamente trabalhos que abordam o uso de bases de dados abertos e conectados e geração de linguagem natural.

3.2 DESCRIÇÃO DOS TRABALHOS

Seguindo o modelo proposto por Vicente et al (2015), os trabalhos relacionados foram divididos de acordo com seu método: sistemas baseados em *templates*, sistemas baseados em técnicas de aprendizado de máquina, e sistemas híbridos.

3.2.1 Sistemas baseados em *templates*

Cojocaru e Trausan-Matu (2015), utilizando como base uma ontologia OWL de vinhos, desenvolveram uma aplicação que realiza a geração de fragmentos de textos através de estruturas semânticas. O objetivo proposto pelos autores foi transformar conhecimento e informações da ontologia em linguagem natural de maneira concisa e expressiva com ajuda da Teoria da Estrutura Retórica (TER). A TER é uma teoria da linguística, que utiliza marcadores de discursos para enfatizar relações entre sentenças (MANN e THOMPSON, 1987). O uso da TER também é o grande diferencial do trabalho dos autores, pois trabalhos anteriores com o mesmo objetivo relataram como resultados sentenças únicas e deficientes em expressividade ao apresentar redundância e/ou sentenças com baixa coesão. Um exemplo deste tipo de trabalho é o OntoVerbal (LIANG et al, 2012), um plugin para o software Protégé, que gera frases em linguagem natural para axiomas de uma classe dentro de uma ontologia.

Cojocaru e Trausan-Matu (2015) finalizaram sua pesquisa, destacando a necessidade de uma técnica para avaliar o resultado da aplicação proposta, pois eles

tiveram que utilizar uma avaliação com usuários para identificar o quão fácil e útil foi o texto gerado pelo sistema gerador de textos desenvolvido.

Atualmente sistemas de perguntas e respostas a partir de bases de dados abertos e conectados baseiam-se somente em responder perguntas a partir de consultas geradas com a linguagem SPARQL. Com objetivo de melhorar a resposta para estes questionamentos, Perera, Nand e Naeem (2017), apresentaram uma nova maneira de construir e mostrar as respostas das perguntas questionadas a sistemas de perguntas e respostas. A abordagem dos autores utiliza a estrutura linguística da pergunta de origem e o fato representado em uma base de conhecimento para construir a resposta que emula o jeito humano de responder uma pergunta. O núcleo da pesquisa dos autores se baseia na extração de padrões de árvore de dependências de questões e utiliza estes padrões na construção das respostas. Primeiramente, o sistema foi avaliado do ponto de vista de acurácia linguística. Após, uma avaliação humana para pontuar as respostas de acordo com a naturalidade de leitura. Depois, uma análise de viabilidade utilizando métricas automáticas para avaliar as respostas geradas contra uma referência a uma resposta criada por um humano. Após essas avaliações, os autores concluíram que a abordagem proposta conseguiu gerar respostas com quocientes de acurácia linguística e leitura natural de mais de 70%.

O trabalho de Perera, Nand e Naeem (2017) se assemelha ao apresentado nesta proposta por utilizar recursos linguísticos e por utilizar *templates* para construção das sentenças. No entanto, ele gera apenas respostas para questionamentos previamente apresentados ao sistema, tirando proveito da construção gramatical da pergunta apresentada em linguagem natural e não utiliza as propriedades RDF para auxiliar na geração do texto.

Mais recentemente, Moussallem et al. (2018) buscaram gerar linguagem natural no idioma português brasileiro a partir de RDF, pois ainda naquele momento não havia nenhum trabalho de geração de linguagem para este idioma. Pelo fato de a língua portuguesa conter uma série de regras morfológicas, os autores optaram por criar uma abordagem a partir de regras e *templates*. Para lexicalização de classes e instâncias, propriedades e objetos, o método dos autores se baseia na propriedade *rdf:label*. Para cada caso, no entanto, algumas peculiaridades do idioma são tratadas de maneira diferente, o que os autores consideram como a principal contribuição de sua pesquisa.

Para a avaliação foram utilizadas pessoas que foram divididas entre usuários experientes e não experientes. Os experientes foram pessoas que tem no mínimo mestrado na área de PLN e SW. Ambos os grupos continham somente nativos na língua portuguesa. Para o grupo de experientes, um questionário contendo perguntas sobre adequação e fluência do texto gerado foi utilizado. Para o grupo de não experientes, foram avaliados os aspectos de clareza e fluência dos textos. Os resultados indicaram que a abordagem foi capaz de capturar e representar a informação de forma correta e também fluente de forma suficiente para leitura humana.

O trabalho dos autores se assemelha ao apresentado nesta proposta por ser baseado em *templates* e regras e por utilizar algumas propriedades das triplas das ontologias (*type* e *label*). No entanto, se diferencia por ter sido desenvolvido apenas para o idioma português e por não ter nenhum apoio de linguistas para construção e avaliação dos *templates*. Além disso, foram utilizados somente três *templates*. O trabalho apresentado nesta proposta, apenas para geração de sentenças, conta com mais de 10 *templates* para construção de sentenças afirmativas e interrogativas.

3.2.2 Sistemas baseados em aprendizagem de máquina

Vougiouklis et al. (2017) exploraram o problema de gerar sumários em linguagem natural para a WEB Semântica, pois a maioria das pessoas não interagem diretamente com ela, precisando de alguma interface caso não tenham a expertise necessária para entendimento da tecnologia. Modelos estatísticos para geração de linguagem natural usando redes neurais foram treinados e avaliados a partir de dois corpora baseados na DBPedia e na WikiData.

O resultado do trabalho foi avaliado de forma automática por conjuntos de métricas e também manualmente por pessoas. As métricas para avaliação automática indicaram, por exemplo, o quanto o modelo aprendeu a partir de seu objeto de treino e também quão perto o texto gerado chegou do sumário original da Wikipedia. A avaliação humana ponderou sobre dois critérios: fluência e número de fatos. Na avaliação automática, os resultados foram inferiores aos que são geralmente reportados por tarefas de tradução de máquina. No entanto, os autores comentam que o resultado deve ser um indicativo de quão bem o modelo pode gerar um sumário textual de triplas correspondentes. O resultado da avaliação humana ficou alinhado

com o resultado da avaliação automática. De modo geral o texto foi avaliado com altas notas para fluência, uma vez que o modelo conseguiu gerar sentenças com gramática e ortografia correta. Com isso, os autores concluíram afirmando que conseguiram criar um modelo para geração de linguagem natural sem a utilização de *templates* e que o modelo pode ser aplicado a uma grande variedade de domínios.

O trabalho de Vougiouklis et al. (2017) se assemelha ao descrito nesta proposta por gerar linguagem natural a partir de bases de dados abertos e conectados, mas se diferencia por utilizar técnicas de aprendizado de máquina minerando textos para encontrar semelhanças entre as triplas e seu conteúdo.

Kaffee et al. (2018) buscaram utilizar as informações estruturadas no WikiData para gerar sumários para os idiomas árabe e esperanto, pois dado que na Wikipedia existem mais de 280 línguas, a maioria dos artigos são escritos em idiomas específicos. Logo, nativos dos idiomas que não têm muito material disponibilizado têm acesso limitado ao seu conteúdo. Para alcançar seu objetivo, os autores utilizaram uma rede neural para gerar as sentenças. Foi utilizada uma arquitetura *Feed-forward* para o *enconding* e uma Rede Neural Recorrente que usa GRUs para *decoder*.

Com o apoio das comunidades destes dois idiomas na Wikipedia, os autores avaliaram a qualidade e a utilidade dos sumários gerados. A avaliação começa medindo o quão próximo o texto gerado fica dos sumários da Wikipedia. Após, um conjunto de métricas automatizadas foram utilizadas. Além disso, os autores também compararam a sua abordagem com outras duas: tradução de máquina e geração de texto baseada em *templates*. Após as avaliações, eles concluíram que o método deles é melhor em gerar sumários do que os demais métodos comparados.

A abordagem de Kaffee et al (2018) se assemelha ao método desta proposta apenas por tentar lexicalizar conteúdo de bases de dados abertos e conectados, neste caso, WikiData. No entanto, se diferencia ao utilizar técnicas de aprendizado de máquina e descartar as informações contidas nas propriedades do RDF para auxiliar na lexicalização.

3.2.3 Sistemas híbridos

Em Duma e Klein (2013), foi proposta uma arquitetura para sistemas de geração de linguagem natural através de dados abertos e ligados que aprende, de forma automática, padrões de frases e planejamento de textos a partir de dados RDF,

corpus paralelo e documentos textuais. O objetivo de comunicação da proposta dos autores era descrever fatos sobre as entidades no estilo Wikipedia. O método proposto pelos autores extrai *templates* através da mineração de textos em conjunto com a análise de grafos RDF que contêm entidades das sentenças encontradas no processo de mineração.

Os resultados do trabalho proposto pelos autores foram avaliados em comparação com um sistema e um texto escrito por um humano a partir das mesmas informações. O sistema de comparação utilizado gerava sentenças únicas para cada tripla RDF a partir de uma análise superficial das palavras do predicado. A arquitetura proposta se mostrou superior ao sistema de frases únicas, mas perdeu em comparação ao texto gerado por humanos. Desta forma, os autores concluíram que sistemas como o proposto podem ser construídos e treinados a partir de um corpus paralelo.

Perera e Nand (2015) exploraram dados abertos e conectados e recursos léxico-semânticos para gerar texto em linguagem natural. Eles construíram um *framework* que gera padrões que podem ser utilizados para lexicalizar uma tripla. O resultado, após a aplicação do padrão, é uma sentença que contém o sujeito e o objeto da tripla na sentença. Algumas vezes, ela também pode conter o predicado. Eles empregaram a técnica de Extração de Informações em uma coleção de sentenças relacionadas aos dados abertos e conectados em conjunto com anotações léxico-semânticas. O *framework* denominado de OpenIE (ANGELI et al., 2015) foi utilizado para extrair relações presentes em textos que podem ser então convertidos para padrões de lexicalização. Além disso, paralelamente, o *framework* busca por padrões de lexicalização com base em *templates* para verbos utilizando VerbNet e WordNet. Os padrões identificados são classificados e categorizados de acordo com a hierarquia de classes da DBPedia.

Os autores avaliaram os padrões sobre dois fatores básicos: assertividade sintática e reuso. A primeira, referindo-se à aderência do padrão à uma representação sintática adequada e coerência do padrão (padrão conciso e coerente). O segundo critério de avaliação identifica se o padrão pode ser generalizado para utilizar em outras triplas que contenham o mesmo conhecimento. Também foi utilizada a técnica MRR (*Mean Reciprocal Rank*) para avaliação, pois o sistema faz um ranqueamento das sentenças geradas (CRASWELL, 2009). Segundo os autores, baseado nos resultados encontrados, a abordagem proposta apresentou um bom desempenho na

derivação de sentenças em linguagem natural por gerar um número justo de padrões de lexicalização válidos e com uma alta precisão para triplas de bases de dados abertos e conectados.

Recentemente, Biran e McKeown (2017) expressaram e analisaram dois grandes problemas com a representação de dados RDF. O primeiro refere-se à falta de informações importantes sobre uma entidade (histórico e contexto). O segundo refere-se ao problema dos dados RDF cruzarem diversos domínios de conhecimento, o que apresenta uma dificuldade ao lidar com domínios específicos na geração de texto. Desta forma, os autores apresentaram um framework genérico de geração de meta-sistemas para aplicações RDF que geram descrição de entidades. O framework se baseia no conceito híbrido de conceito-para-texto (C2T) e texto-para-texto (T2T): parte do texto é composto a partir de dados estruturados de acordo com uma estrutura genérica, enquanto outra parte do texto é extraída de um corpus de domínio. Além disso, são utilizados métodos para extrair paráfrases e modelos de discurso para refinar o texto final.

Os autores avaliaram o sistema a partir de um experimento com profissionais. Cada participante tinha 4 critérios para avaliação: relevância da informação no conteúdo do texto; ordem das sentenças e parágrafos; estilo do texto; e satisfação em relação ao texto apresentado. A partir do resultado da avaliação dos autores, eles concluíram que a proposta híbrida em conjunto com a adaptação de domínio de conhecimento resultou em descrições textuais satisfatórias.

O experimento dos autores se assemelha ao descrito nesta proposta por partir da assimilação de possessão entre os elementos de uma tripla, pois alguns *templates* gerados pelos linguistas também apontou construções textuais de possessão entre os elementos. No entanto, para lexicalização das triplas, não são consultadas as propriedades RDF. Além disso, o framework proposto minera um corpus auxiliar para encontrar sentenças que contenham os termos da tripla e variam de acordo com o domínio do conhecimento contido na tripla RDF.

3.3 Comparativo

De fato, todos os trabalhos mencionados apresentaram êxito na geração de linguagem natural a partir de dados abertos e conectados aplicando diferentes técnicas e respondem os questionamentos levantados na fase de planejamento da

revisão da literatura. Desta forma, a partir dos trabalhos apresentados, é possível concluir que há formas de gerar linguagem natural a partir de dados abertos e conectados. As principais formas puderam ser divididas em arquiteturas e métodos apoiados em *templates*, em técnicas de aprendizado de máquina e arquiteturas híbridas. Além disso, os trabalhos relacionados relatam diferentes aplicações para geração de linguagem a partir de bases de dados abertos e conectados. Eles podem ser utilizados para gerarem e responderem perguntas em linguagem natural em sistemas de perguntas e respostas, por exemplo. Também podem ser utilizados para gerar descritivos de conceitos presentes nas bases de dados abertos e conectados, entre outras aplicações.

Os primeiros trabalhos descritos, de forma geral, se baseiam em *templates*, sejam elas manualmente construídas ou geradas a partir de técnicas de aprendizagem de máquina. Alguns trabalhos ainda optaram pelos dois modelos, projetando sistemas híbridos de geração de linguagem natural.

O Quadro 1 organiza os principais aspectos destacados a partir deste estudo de trabalhos relacionados. Ele auxilia na visualização dos pontos onde esta pesquisa se diferencia em relação aos demais trabalhos estudados. O estudo realizado utilizou os seguintes atributos para apoiar a comparação entre os trabalhos: análise do método utilizado, tipo de avaliação de resultados realizada, aspectos do uso do padrão RDF e léxicos. A seguir estes atributos e sua pertinência são descritos.

Com relação ao método utilizado, conforme descrito anteriormente na seção de fundamentação teórica, algoritmos de geração de linguagem a partir de BDAC podem ser divididos em: algoritmos baseados em templates, algoritmos baseados em técnicas de aprendizado de máquina e algoritmos híbridos. Com relação ao procedimento de avaliação de resultados, a partir de todos os trabalhos analisados, notou-se a divisão das técnicas de avaliação em dois agrupamentos: validações automáticas do texto gerado e avaliação por pessoas sem conhecimento da área da linguística. Assim, os trabalhos foram agrupados desta maneira, destacando que nenhum dos trabalhos relacionados teve apoio de profissionais da linguística para avaliação dos resultados de seus trabalhos.

Sendo um dos diferenciais deste trabalho, o uso das propriedades RDF foi uma das dimensões utilizadas para comparação. Entre os poucos trabalhos que exploram essas propriedades, eles estavam limitados às propriedades *type* e *label* dos conceitos presentes nas BDAC. O uso extensivo das propriedades RDF pode fornecer

pistas semânticas para construção de sentenças, o que não foi explorado pelos trabalhos relacionados com essas duas únicas propriedades RDF aproveitadas.

Como última dimensão de comparação, entende-se que o uso de léxicos para apoiar na construção das sentenças é um recurso diferencial para geração de frases mais naturais e diversificadas para as triplas RDF.

Quadro 1 – Análise de trabalhos relacionados.

Trabalho relacionado	Quanto ao método			Quanto à avaliação			Quanto ao RDF			Léxicos	
	Templates	Apren. de máquina	Híbrido	Auto.	Pessoas não-técnica	Ling.	Prop. *label*	Prop. *type*	Demais propr.	Sinôni-mos	Comb. de palavras
BIRAN, O.; MCKEOWN, K. (2017)			X		X						
COJOCARU, D. A.; TRAUSAN-MATU, S. (2015)	X										
DUMA, D.; KLEIN, E. (2013)			X		X						
KAFFEE, L. A. et al (2018)		X		X	X						
LIANG, S. F. et al (2012)	X						X				
MOUSSALLEM, D. et al (2018)	X				X		X	X			
PERERA, R.; NAND, P. (2015)			X	X	X					X	
PERERA, R.; NAND, P.; NAEEM, A. (2017)	X			X	X						
VOUGIOUKLIS, P. et al (2017)		X		X							
Este trabalho (2018)	X				X	X	X	X	X	X	X

Fonte: Elaborado pelo autor.

Como pode ser observado no Quadro 1, o trabalho apresentado nesta proposta se assemelha aos trabalhos que se baseiam em *templates*. Porém, também se diferencia destes por utilizar *templates* geradas em conjunto com um grupo de linguistas. Com a parceria dos especialistas da linguística tentou se obter *templates* que modelassem frases afirmativas e interrogativas mais naturais e corretas

gramaticalmente, pois entende-se que os profissionais são mais qualificados neste quesito por compreender os domínios da linguagem e suas teorias.

Os *templates* foram construídos a partir de um corpus composto com diversas sentenças geradas pelos profissionais da área onde o objetivo era obter frases que soassem naturais e que fossem gramaticalmente corretas. Para tanto, conforme mencionado anteriormente, os linguistas utilizaram as formas lógicas das linguagens ativa e passiva para gerar diferentes frases, além de explorarem diferentes pronomes e diferentes tipos de pontuação.

Além disso, os poucos trabalhos que geram linguagem a partir de *templates*, tem um número limitado de *templates* (entre três e cinco *templates*). No caso do trabalho desta proposta, inicialmente já foram gerados mais de dez *templates*, entre *templates* para sentenças afirmativas e interrogativas.

Também, diferentemente dos demais, conforme Quadro 1, este trabalho tem como premissa, além da avaliação de pessoas nativas do idioma do texto gerado pelo sistema, a avaliação de especialistas da área da linguística para garantir que as sentenças, tanto afirmativas quanto interrogativas, tenham um alto grau de precisão gramatical e naturalidade.

Ainda observando as dimensões de comparação do Quadro 1, observa-se que o maior diferencial a destacar neste trabalho é o uso das definições presentes no documento descritivo sobre o RDF do W3C. Como o RDF é considerado o *framework* padrão para intercâmbio de dados na WEB, algumas de suas propriedades podem conter informações importantes para auxiliar no processo de lexicalização de uma tripla RDF. Alguns trabalhos citados utilizam uma ou duas propriedades, mas não exploram profundamente o potencial das informações contidas nessas e em outras propriedades.

Também, para dar flexibilidade para geração de sentenças, diminuindo a probabilidade das informações de uma tripla RDF sempre resultarem em uma mesma sentença, o algoritmo proposto neste trabalho faz uso de um léxico de sinônimos e um léxico de combinações de palavras. Em outras palavras, ao trocar elementos-chaves das sentenças por sinônimos, diminui a possibilidade de gerar frases repetidas para um determinado conjunto de informações e se diferencia dos demais trabalhos ao utilizar esses recursos linguísticos (Quadro 1).

Por fim, observa-se no quadro comparativo pontos que os demais trabalhos não exploram completamente para geração de linguagem a partir de BDAC. Portanto,

este trabalho visa explorar as lacunas presentes conforme destacado anteriormente nas dimensões comparativas.

4 ALGORITMO

Neste capítulo será descrito o algoritmo proposto para a geração de frases curtas em linguagem natural a partir do uso de informações estruturadas em formato RDF disponíveis em Dados Abertos e Conectados.

De forma resumida, o algoritmo inicia sua operação a partir da escolha de uma entidade sobre o qual se deseja construir frases. O passo seguinte consiste na busca automática de informações sobre este conceito. Isso é realizado a partir de consultas no conteúdo disponível em uma ou mais bases de dados abertos e conectados, considerando que este conteúdo esteja estruturado com o padrão RDF. Com as informações obtidas nesta consulta, o algoritmo utiliza recursos linguísticos para gerar frases afirmativas e interrogativas em linguagem natural.

Com o objetivo de melhorar a qualidade e diversidade das sentenças geradas, o algoritmo está estruturado com base em uma série de premissas e de uso de recursos complementares. Estes recursos representam os principais diferenciais deste algoritmo em relação ao conjunto de trabalhos relacionados estudados. O primeiro recurso a destacar é o uso extensivo das informações de propriedades dos elementos de triplas RDF. O segundo é o uso de *templates* linguísticos construídos por um grupo de profissionais da área da linguística. Um terceiro consiste na utilização de léxicos para obtenção de sinônimos e para flexibilizar a combinação de palavras geradas. Seus detalhes serão comentados a seguir, em seções específicas.

A primeira seção deste capítulo descreve, portanto, a utilização das propriedades dos elementos de um tripla RDF. Na sequência, será descrita a elaboração dos *templates* linguísticos. Depois, serão descritos os dois componentes adicionais do algoritmo que são consultados para melhorar a qualidade de sentenças geradas: um léxico de sinônimos e um léxico de combinações de palavras. Por fim, o algoritmo é descrito.

4.1 INFORMAÇÕES CONTIDAS NA TRIPLA RDF

A especificação da modelagem de dados RDF descreve o conceito de propriedades RDF como a relação entre um sujeito e um objeto (RDF, 2014a). Dentro da especificação também são descritas propriedades padrões que podem ser utilizadas em uma modelagem de dados.

O Quadro 2 apresenta as propriedades RDF que são utilizadas pelo algoritmo com suas respectivas descrições. As propriedades listadas fornecem informações significativas para escolha de partes de uma sentença objetivando alto grau de naturalidade e coerência gramatical, pois cada propriedade está claramente associada a um determinado contexto semântico. Somente a propriedade *rdfs:subClassOf* descrita na definição de RDF1.1 da W3C (RDF, 2014a) não foi objeto de estudo deste trabalho por limitação de tempo. No entanto, ela, em conjunto com as demais, pode ser plenamente estudada em trabalhos futuros.

Para exemplificar, o Quadro 2 ilustra a descrição da propriedade *label* como associada a uma descrição geral do nome do recurso sendo representado. Já, por sua vez, a propriedade *range* claramente está associada com uma faixa de valores que podem ser associados com recursos que esta descreve. O Quadro 2, na coluna 'Utilizado na Academia' também indica, dentre essas propriedades, as que foram possíveis identificar nos trabalhos relacionados nesta pesquisa.

Quadro 2 – Propriedades RDF.

Propriedade	**Descrição**	**Utilizado na Academia**
label	Nome do recurso em linguagem natural	Sim
range	Determina que o valor de uma propriedade é instância de uma ou mais classes	Não
type	Usado para identificar o tipo do recurso	Sim
domain	Determina que qualquer recurso que tem a propriedade é uma instância de uma ou mais classes	Não
comment	Descrição em linguagem natural de um recurso	Não
subpropertyOf	Indica que todos os recursos relacionados por uma propriedade também estão relacionados por outra.	Sim

Fonte: Elaborado pelo autor.

São essas propriedades padronizadas que o algoritmo descrito neste trabalho utiliza para obter maiores informações sobre o conhecimento representado em uma tripla e, assim, gerar texto em linguagem natural. Alguns dos trabalhos relacionados utilizam algumas dessas propriedades para lexicalização e outras funções dentro de seus algoritmos de geração de linguagem. No entanto, este trabalho se diferencia dos

demais pelo uso extensivo destas propriedades e, desta forma, contribui com o estudo do uso de mais propriedades RDF na geração de linguagem natural.

Nas próximas subseções serão descritos detalhes de como essas propriedades podem ajudar na composição de uma sentença em linguagem natural.

4.1.1 Uso da propriedade *rdfs:label*

O uso da propriedade *rdfs:label* auxilia na lexicalização de um determinado recurso. No contexto de uma tripla RDF, todos os três membros devem conter a propriedade *rdfs:label*. O conteúdo da propriedade corresponde, na grande maioria dos casos, a palavra em linguagem natural que representa o recurso. Além disso, essa propriedade pode conter múltiplos valores que correspondem a palavra em outros idiomas (RDF, 2014b).

A partir desta premissa, essa propriedade é usada para substituir os substantivos que irão representar sujeitos, predicados e objetos em uma sentença afirmativa a partir de uma tripla RDF. Em uma sentença interrogativa, como o objeto da tripla é objeto da pergunta, essa propriedade fornece o substantivo que será usado para representar o sujeito e o predicado da sentença afirmativa.

Importante destacar também que, para ambos tipos de sentença, afirmativa e interrogativa, o valor da propriedade *rdfs:label* também auxilia na busca pelo artigo definido gramaticalmente correto para acompanhá-la nas sentenças que serão geradas. Além disso, essa propriedade já é utilizada por outros trabalhos com o mesmo objetivo (LIANG et al, 2012; MOUSSALLEM et al, 2018).

4.1.2 Uso da propriedade *rdfs:range*

Esta propriedade, quando presente, indica o tipo do objeto de um tripla RDF. No algoritmo descrito neste trabalho, ela acaba não sendo utilizada para geração de sentenças afirmativas. No entanto, tem papel fundamental na composição de sentenças interrogativas.

Os pronomes interrogativos têm algumas regras para seu emprego em composições interrogativas. Desta forma, o algoritmo prevê uma etapa de pré-configuração antes de sua execução com o objetivo de relacionar as classes da ontologia com palavras da língua inglesa para correto emprego dos pronomes.

Partindo deste princípio, a indicação do tipo do objeto pela propriedade *rdfs:range* é uma pista essencial para definição do pronome interrogativo que será utilizado na composição da sentença interrogativa.

Por exemplo, na etapa de pré-configuração, a classe Pessoa da DBPedia foi relacionada com o substantivo 'Pessoa'. Desta forma, o algoritmo acaba escolhendo o pronome '*Who'* como pronome interrogativo da sentença.

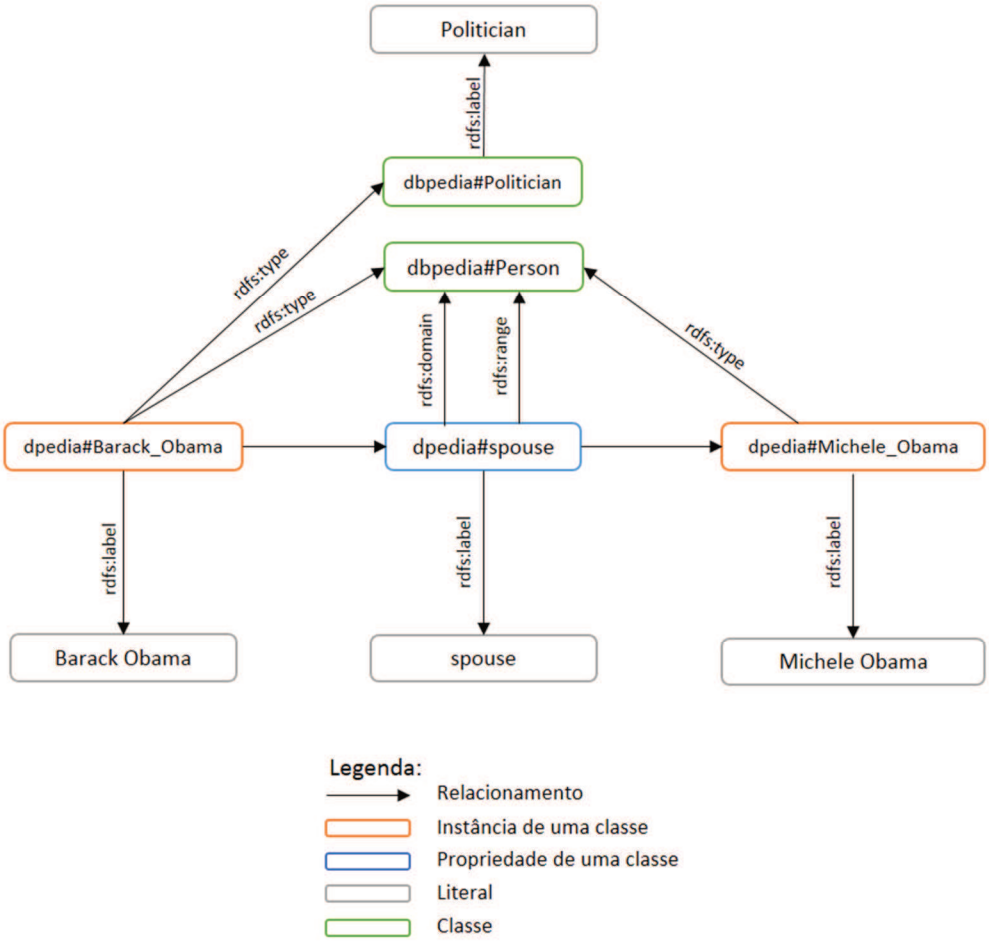
No entanto, até o momento, não há referência do uso desta propriedade como auxílio na geração de linguagem natural dentre os trabalhos encontrados a partir da revisão bibliográfica executada no início desta pesquisa.

4.1.3 Uso da propriedade *rdf:type*

Esta propriedade também já foi utilizada por outros pesquisadores em algoritmos de geração de linguagem natural. Dentro do algoritmo descrito neste trabalho ela é usada para indicar o tipo de um recurso. Ela pode apontar para uma classe da ontologia ou uma classe da própria definição do RDF (RDF, 2014a).

Partindo deste princípio, o uso dela é diversificado. Quando aponta para uma ou mais classes da ontologia, o *rdfs:label* dessas classes podem ser utilizadas em sentença que contém orações explicativas. No caso do subgrafo apresentado na Figura 3 que contempla algumas relações do recurso Barack Obama. Seguindo o princípio descrito, a partir do subgrafo, é possível gerar a sentença: "*Barack Obama is a politician, whose spouse is Michele Obama*".

Figura 3 – Exemplo de um subgrafo RDF de um relacionamento.



Fonte: Elaborado pelo autor.

Além disso, quando *rdfs:type* aponta para um tipo de dado que não seja uma classe, métodos específicos vão tratar cada um dos tipos de dados suportados pelo algoritmo (literais, datas, números inteiros, números flutuantes, etc.).

4.1.4 Uso da propriedade *rdfs:comment*

O valor desta propriedade também foi descoberto após execução da primeira avaliação. Descobriu-se que, exclusivamente quando conectado com a DBPedia como base de dados, essa propriedade pode indicar que uma determinada relação é de uso reservado da DBPedia, não sendo relevante para geração de sentenças em linguagem natural. Desta forma, essa propriedade, quando presente, se indicar que a

relação é de uso da exclusivo da DBPedia, não seleciona o relacionamento para geração de linguagem natural.

4.1.5 Uso da propriedade *rdfs:subPropertyOf*

Essa propriedade indica que todos os recursos relacionados por uma propriedade também são relacionados por outro. Ela tem um uso bem específico dentro do algoritmo na indicação que um recurso é membro de um conjunto de outros elementos.

Por exemplo, para o relacionamento *Sujeito* → *board* → *Objeto*, há a indicação que o sujeito deste relacionamento é um dos membros do objeto. Desta forma, é possível gerar diferentes sentenças para representar essa informação. Os *templates* que formam estas sentenças são listadas no Apêndice A juntamente com os *templates* que geram outras sentenças.

4.1.6 Exemplo prático do uso das propriedades RDF

Exemplificando um pouco melhor o uso destas propriedades, para composição de uma sentença em linguagem natural, observa-se na Figura 3 uma situação mais detalhada, na qual, dada a tripla expressa pelo relacionamento entre as instâncias “Barack Obama” e “Michele Obama”, a análise dos aspectos de propriedades RDF possibilita obter uma série de informações para geração de linguagem. A partir da propriedade *label* do predicado da tripla é possível identificar o nome da relação entre as duas pessoas apontadas. Também é possível identificar, a partir das propriedades *domain* e *range*, que este conceito está associado com o conceito de pessoa, no caso com o conceito “*dbpedia:Person*”. Portanto, como a relação analisada (“*dbpedia:spouse*”) possui a propriedade *range* associada com o conceito pessoa, no caso de uma sentença interrogativa, isto seria determinante para que o pronome escolhido seja “Quem”, ou a sua tradução para inglês, ‘*Who’*.

4.2 *TEMPLATES*

O algoritmo tem como objetivo gerar sentenças curtas afirmativas e interrogativas a partir de dados abertos e conectados. Para tanto, foi adotada uma

abordagem que utiliza um conjunto de *templates* com marcações que poderão ser substituídas por elementos do subgrafo RDF em torno de um relacionamento entre sujeito e objeto para a geração de sentenças em linguagem natural.

Um grupo de três especialistas da área da linguística foi convidado para elaborar um conjunto de sentenças para 12 diferentes triplas RDF extraídas da DBPedia. Metade delas para a classe com o maior número de instâncias na DBPedia, a classe Pessoa. A outra metade para a segunda classe com maior número de entidades descritas na DBPedia, a classe Lugar (ABOUT, 2017). Os relacionamentos de cada entidade dessas classes foram escolhidos de acordo com o tipo de dados do seu objeto, pretendendo-se com essa seleção gerar diferentes *templates* para a maioria dos tipos de dados presentes na DBPedia.

A partir de instâncias escolhidas por conveniência, os especialistas foram convidados a gerar cinco afirmações e cinco questionamentos para cada um dos relacionamentos. No total, foram geradas cento e oitenta sentenças. A partir das sentenças, o grupo de linguistas, em conjunto com o autor deste trabalho, construíram quinze *templates*.

Para construção das sentenças afirmativas e interrogativas geradas pelos especialistas da área da linguística, foram utilizadas as informações presentes em uma tripla RDF para construir sentenças que soassem naturais e corretas gramaticalmente. Para isso, diferentes formas lógicas (por exemplo, as linguagens ativa e passiva) foram utilizadas e pronomes comumente utilizados no contexto de frases foram também incluídos, além da exploração de diferentes tipos de pontuação.

Nessas sentenças, observou-se quais informações presentes no grafo RDF estavam descritas como um elemento sintático da frase. A partir dessas observações, foram construídos sete *templates* para sentenças afirmativas para objetos com tipo de dados numérico, textual e temporal. Para sentenças interrogativas, foram construídos oito *templates* para objetos dos mesmos tipos.

Em cada *template* foram utilizadas marcações dos elementos que podem ser substituídos dinamicamente por informações contidas na tripla. O Quadro 3 descreve as marcações utilizadas nos *templates*. A lista completa de *templates* pode ser visualizada no Apêndice A.

Quadro 3 – Marcações utilizadas nos *templates*.

Marcador	**Descrição**
S	Propriedade *label* do sujeito da tripla
P	Propriedade *label* do predicado da tripla
PV	Um verbo que pode substituir o *label* do predicado da tripla
O	Propriedade *label* do objeto da tripla
OT	Propriedade *type* do objeto da tripla
PRON	Pronome interrogativo
AP	Apóstrofo
TIME	Representação textual de uma data
VTB	Verbo *to be*
ST	Classe do sujeito
A	Artigo
VHA	Verbo *to have*
PRPOSS	Pronome possessivo

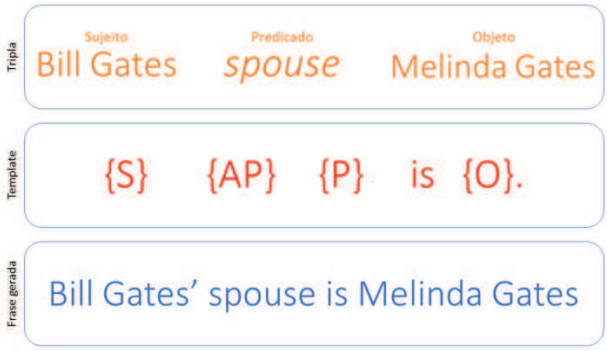
Fonte: Elaborado pelo autor.

Neste primeiro estudo em conjunto com os linguistas, um total de treze marcadores foram gerados (Quadro 3) a partir das sentenças construídas. Destaca-se que o algoritmo definido suporta a possibilidade de expansão do seu funcionamento baseada em um conjunto maior de sentenças. Deste modo, novos marcadores podem ser identificados para gerar frases mais complexas em trabalhos futuros.

Cada marcador representa um elemento em uma sentença. Ele pode ser sintático (sujeito, predicado, objeto ou pronome) ou ainda um sinal gráfico, como o apóstrofo. Além disso, há também um marcador criado para auxiliar na formatação da informação temporal, para sentenças que tenham seus complementos sendo uma data. A Figura 4 representa uma sentença gerada a partir das informações de uma tripla que foram substituídas em um *template.* Nela pode ser observado como os marcadores descritos no Quadro 3 são utilizados para integrar os elementos da tripla exibida na Figura 4 e assim formar uma frase. O primeiro elemento da Figura 4 (o elemento “tripla”) descreve o resultado obtido em uma consulta a uma BDAC, a partir da qual pode ser recuperado, para cada elemento da tripla, sua propriedade *label*. Ou seja, o *label* do sujeito da tripla é o nome “Bill Gates”, o *label* do predicado da tripla é o termo “*spouse*” e o *label* do objeto da tripla, ou seja, o nome “Melinda Gates”. O segundo elemento da Figura 4 ilustra um dos *templates* utilizados neste trabalho. Este *template* é composto pelos marcadores “S”, “AP”, “P” e “O”, que indicam respectivamente a propriedade *label* do sujeito da tripla, o caractere apóstrofo, a

propriedade *label* do predicado da tripla e a propriedade *label* do objeto da tripla. O elemento final da Figura 4 descreve o resultado da substituição simples dos marcadores pelos elementos recuperados com a consulta da tripla.

Figura 4 – Sentença gerada a partir de uma tripla RDF e um *template.*



Fonte: Elaborado pelo autor.

4.3 FLEXIBILIZAÇÃO E DIVERSIFICAÇÃO NA GERAÇÃO DE SENTENÇAS

O algoritmo prevê o uso de recursos adicionais para aumentar a qualidade do texto gerado para as triplas RDF. O primeiro recurso previsto é um léxico de sinônimos para ampliar o número de possibilidades de geração de diferentes sentenças com os mesmos conceitos e relações. A busca de sinônimos seria utilizada para as palavras das propriedades *label* de cada elemento da tripla. Neste caso, considerando o exemplo da Figura 3, onde o *label* do predicado é '*spouse'*, uma consulta no léxico de sinônimos poderia retornar as palavras: '*husband'*, '*wife'*, '*mate'*, '*partner'*, entre outros.

Também é previsto o uso de um léxico de combinações de palavras. No cenário do algoritmo, existe um marcador que prevê a verbalização da propriedade *label* do predicado (marcador *{PV}* no Quadro 3). Por exemplo, para a palavra '*spouse*' da propriedade *label* do predicado do exemplo da tripla da Figura 3, a busca no léxico de combinações poderia indicar qual é o verbo mais usual para a palavra '*spouse*'. Neste caso, poderia ser obtido o verbo '*marry*'. Deste modo, a consulta a este léxico garante uma maior aproximação dos resultados gerados com os termos usados amplamente na linguagem falada.

Para implementação do algoritmo, no quesito dos léxicos de suporte, utilizou-se o WordNet (MILLER, 1995), o léxico de combinações de palavras "freecollocation.com" e o a API Datamuse para obter a frequência das palavras

(DATAMUSE, 2018). O primeiro conta com aproximadamente 115 mil conjuntos de sinônimos (*synsets*).

O segundo léxico, por sua vez, contém mais de 150 mil combinações de palavras para aproximadamente 9 mil palavras-chave (OXFORD, 2018). Este léxico foi escolhido por indicação dos linguistas que participaram da construção dos *templates,* por tratar-se de uma ferramenta amplamente utilizada no escopo do estudo da língua inglesa.

Já a API para frequência das palavras foi utilizada para consultar a frequência dos sinônimos obtidos a partir da consulta ao WordNet pelo sinônimo das palavras. Isso foi necessário para evitar que palavras pouco usuais fossem escolhidas pelo algoritmo para composição das frases. A escolha de palavras pouco usuais poderia afetar o aspecto natural de uma sentença gerada pelo algoritmo. Esse comportamento foi observado nas avaliações, conforme serão relatados no capítulo de avaliações.

4.4 DESCRIÇÃO DO ALGORITMO E SUAS ETAPAS

Nesta seção é descrito em detalhes o processo do algoritmo proposto, seus principais componentes e o fluxo geral de sua execução.

Todas as frases são geradas com informações existentes em bases de dados abertos e conectados. Estas bases armazenam dados em formato RDF e permitem que sejam realizadas consultas usando a linguagem SPARQL. Portanto pode-se considerar este procedimento de consulta à uma base de dados abertos e conectados como equivalente à consulta em uma ontologia descrita em formato RDF.

O primeiro passo do algoritmo é a obtenção de uma URI que representa o conceito para o qual se deseja gerar frases. Este passo equivale à busca de uma instância na ontologia utilizada como fonte de consulta. Este passo está indicado com o rótulo “1º Passo” na Figura 5. O algoritmo é flexível quanto à escolha da base de dados abertos e conectados a ser empregada, desde que esta seja modelada com o padrão RDF. Essa etapa inicial, onde é realizada a consulta e obtida uma URI com uma instância, tem como objetivo validar se o resultado obtido é uma instância válida.

Após esta validação da URI, todos os relacionamentos da instância são extraídos das bases de dados abertos e conectados em questão. Este procedimento está descrito na Figura 5, com o rótulo “2º Passo”. A partir desta lista de relacionamentos, ocorre um processo de validação. Neste processo, para cada item

da lista busca-se identificar se a tripla atende os critérios para geração de texto, eliminando as triplas que não são válidas. Estes passos estão indicados na Figura 5 com os rótulos “3º Passo” e “4º Passo”. Esses critérios foram identificados a partir da avaliação dos resultados da avaliação inicial desta pesquisa, que é a primeira avaliação que está descrita no capítulo 6. Eles definem se uma tripla é passível de geração de texto pelo algoritmo ou não. Os critérios são os seguintes:

- Validação se o predicado é uma URI;
- Validação se o predicado faz parte das ontologias definidas no sistema;
- Verificação se o predicado contém a propriedade *domain* definida e se o valor desta propriedade é um URI.

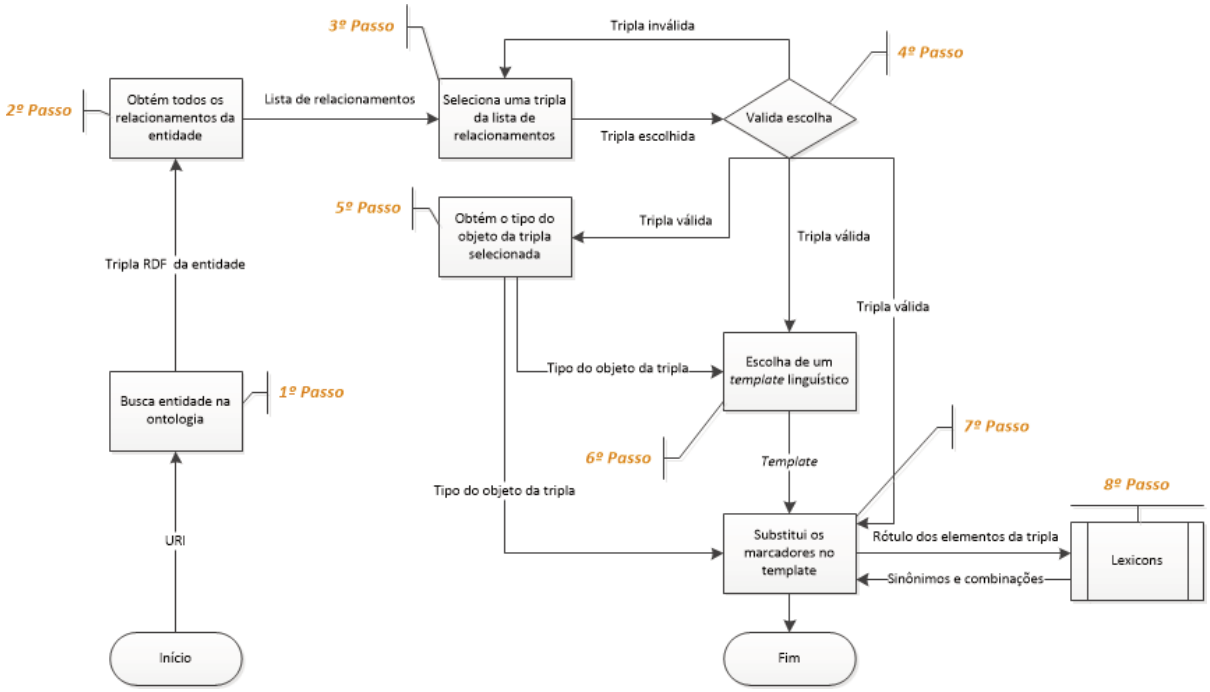
A partir da lista de relacionamento obtida no 3º Passo do algoritmo, o processo de validação só irá finalizar quando encontrar uma tripla adequada para geração da sentença. Logo, se uma tripla não atender os critérios definidos, ela é removida da lista para geração de texto.

É importante destacar que o algoritmo pode gerar mais de uma sentença para uma entidade. Ou seja, ele pode permanecer em um processo de tratamento dos itens da lista de relacionamentos para encontrar todos os relacionamentos válidos.

No 5º passo indicado na Figura 5, para cada tripla que foi validada, segundo os critérios mencionados, deve ser extraído o tipo de objeto da tripla. Desta forma, uma nova busca é executada para obter o tipo do objeto através da propriedade RDF *type*. Essa propriedade é importante para apoiar o preenchimento correto de alguns *templates* de sentenças afirmativas e interrogativas.

Seguindo esta lógica, do 1º passo ao 5º passo, o algoritmo parte para o estabelecimento das informações que irão aparecer nas sentenças. Desta forma, seguindo o modelo abstrato definido por Aires em 2016, até o 5º passo são etapas da fase de planejamento de texto (*text planning)*.

Figura 5 – Fluxograma do algoritmo.



Fonte: Elaborado pelo autor.

Dada uma tripla da lista de relacionamentos da instância, a partir do tipo de dado e propriedades de seus elementos, um conjunto de *templates* linguísticos adequado ao tipo de dados da tripla é escolhido. Dado que há vários *templates* para um determinado tipo de dado, um dos *templates* é escolhido de forma randômica para ser a base da sentença a ser gerada (6º Passo na Figura 5). Esta escolha randômica permite que seja gerada uma diversidade maior de frases. Após a escolha do *template*, ele é utilizado para guiar todas as trocas de elementos, substituindo as marcações previamente definidas por informações das propriedades dos elementos da tripla (7º Passo na Figura 5). É nesse processo de trocas que ocorre a utilização dos léxicos de sinônimos e combinação de palavras para aumentar a diversificação e melhorar a qualidade das sentenças (8º Passo na Figura 5).

Desta forma, a partir do modelo abstrato descrito por Aires em 2016, todas as tarefas do 6º ao último passo deste algoritmo, pertencem a fase de planejamento de sentença (*sentence planning*). O algoritmo não implementa nenhuma tarefa para a fase de síntese do discurso (*speech synthesis*), pois assume-se que os *templates* já contém as regras gramaticais necessárias para que a frase curta esteja sintaticamente e morfologicamente correta.

4.4.1 Exemplo ilustrativo

Dentro deste contexto, por exemplo, dado o URI da instância “Bill Gates”, o primeiro passo do algoritmo é validar se essa instância realmente existe na BDAC. Nesse caso, o algoritmo irá consultar a DBPedia verificando se a instância existe na base. Após essa validação, uma nova consulta será executada para obter todos os relacionamentos dessa instância. A partir desta lista, primeiramente, o algoritmo irá validar cada uma das relações até encontrar uma tripla para gerar uma sentença. Vamos supor que, na lista de relacionamentos da entidade “Barack Obama”, o algoritmo seleciona o relacionamento com o predicado “*spouse*” (Figura 3). Após, com a tripla escolhida, uma nova consulta é disparada para encontrar o tipo do objeto do relacionamento. No caso do nosso exemplo, o objeto do relacionamento onde o sujeito é “Barack Obama” e o predicado é “*spouse*”, é “Michele Obama”. E esse objeto é do mesmo tipo do sujeito, sendo uma entidade da classe “Pessoa” da ontologia da DBPedia. O tipo de dado do objeto, que no caso é um URI apontando para a entidade “Michele Obama”, é avaliado para selecionar o melhor *template* para gerar uma sentença em linguagem natural.

Existem mais de um *template* para cada tipo de dado. A escolha entre os *templates* para determinado tipo de dado é feita randomicamente. A partir do *template*, o processo de substituição ocorre em cada uma das marcações. No cenário de exemplo da Figura 4, vamos supor que o *template* selecionado seja “*{S} {AP} {P} is {O}.*”

Para esse *template*, a marcação *{S}* será substituída pela propriedade *label* do sujeito, no caso “Bill Gates”. A marcação *{AP}* seria substituída pelo sinal de apóstrofo correto para indicar a possessão do sujeito da oração em relação ao predicado, marcado com *{P}.* Para a substituição da marcação *{P}*, um léxico de sinônimos é consultado a partir da propriedade *label* do predicado. Vamos supor que, ao consultar o léxico, uma lista de sinônimos foi retornada. A partir desta lista, o algoritmo escolhe randomicamente um sinônimo para preencher o marcador. Neste caso, vamos supor que ele escolhe a própria palavra “*spouse*”. Por fim, a marcação do objeto da oração representado por *{O}* é substituída pela propriedade *label* do objeto da tripla, pois o tipo de dados do objeto da tripla de exemplo é um URI. A Figura 4 sintetiza o exemplo e mostra a sentença resultante do template aplicado na tripla RDF.

5 IMPLEMENTAÇÃO

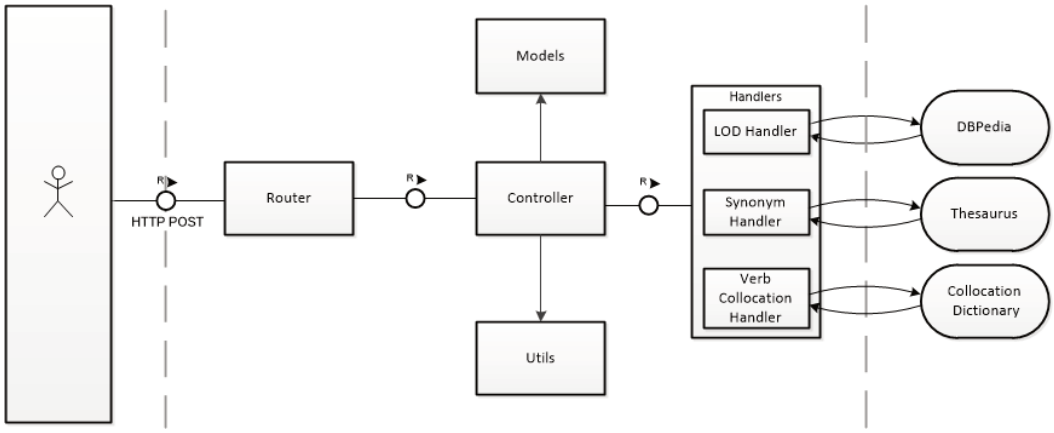
O algoritmo foi implementado para gerar sentenças no idioma inglês americano. Para tanto, a base de dados aberta e conectada utilizada foi o capítulo em inglês da DBPedia. A implementação foi construída para ser modular. Desta forma, novas funcionalidades e regras podem ser adicionadas sem muita dificuldade. Nas próximas subseções, a estrutura e o funcionamento da implementação serão descritos e explicados.

5.1 ESTRUTURA

Para implementação do algoritmo, foi desenvolvida uma aplicação nodeJS utilizando a linguagem JavaScript. Durante a concepção da solução de implementação, foi escolhido projetar a aplicação como um sistema *backend*, pois ele teria como principal objetivo a geração de linguagem natural através de uma entidade de uma base de dados aberta e conectada. Desta forma, outras aplicações poderiam consumir os serviços expostos por essa implementação. Para tanto, a aplicação foi estruturada em camadas. Cada camada possui sua responsabilidade única.

Para o desenvolvimento da aplicação nodeJS, foi utilizado o *framework* Express que entrega os recursos necessários para construir um servidor WEB (EXPRESS, 2017). A partir deste *framework*, a aplicação foi separada em camadas (Figura 6). A primeira camada corresponde às rotas (*routers*). As rotas de cada funcionalidade foram separadas por arquivos. Neste primeiro momento, a aplicação tem apenas duas funcionalidades: geração de sentenças afirmativas e geração sentenças interrogativas. Desta forma, dentro dos arquivos destas funcionalidades, rotas para cada um dos métodos HTTP foram criadas. Atualmente, para simplificação da implementação do algoritmo, somente os métodos POST de cada rota estão respondendo requisições HTTP.

Figura 6 – Componentes da implementação do algoritmo utilizando TAM (TECHNICAL, 2013).



Fonte: Elaborado pelo autor.

Cada uma das duas rotas da aplicação recebe uma chamada com dois parâmetros conforme prevê o algoritmo: um parâmetro contendo a URI da entidade de uma base de dados aberta e conectada e um número indicando a quantidade de sentenças que deverão ser geradas para a entidade em questão. Além de ser a porta de entrada da requisição, na rota também são realizadas algumas validações nos parâmetros enviados nas chamadas: verificação de nulidade e tipagem correta dos parâmetros. Caso as validações sejam feitas com sucesso, a rota é responsável por interagir com a próxima camada, a camada controladora (*controller*).

As controladoras são responsáveis por orquestrar todo o processo de construção das sentenças assim que receber uma requisição válida da rota. As controladoras interagem diretamente com outras controladoras para atender uma requisição. Cada controladora é responsável por um domínio de conhecimento.

No caso da implementação do algoritmo, existem três controladoras. Uma controladora para orquestrar as funções referentes às entidades da DBPedia (*EntityController)*, uma controladora responsável pelo escopo das relações entre as entidades da DBPedia (*RelationController)* e, por fim, uma controladora para comandar a construção das sentenças, tanto afirmativas, quanto interrogativas (*SentenceController)*. Essas controladoras interagem com os modelos e com os manipuladores de tarefas específicas.

Os modelos (*Models)*, dentro da implementação deste algoritmo, correspondem a representações de entidades. Modelos não armazenam somente dados, mas também têm a responsabilidade de controlar comportamentos. Mais

especificamente, o modelo *SentenceModel* é responsável por escolher um *template* e substituir os marcadores existentes pelas informações da entidade e suas relações com outras entidades.

Além disso, os manipuladores (*handlers*) são funções que têm responsabilidades de tarefas específicas que não pertencem a um modelo. Nesta implementação, existem manipuladores para tratamento das conexões com a base de dados aberta e conectada (*SparqlHandler)*, manipuladores de erros (*ErrorHandler*) e de conexão com os léxicos (*HTTPHandler*).

Além do que foi apontado, a implementação conta com uma última camada que contém as funções utilitárias e arquivos de configuração. Dentro desta camada também se encontram todos os *templates* para sentenças afirmativas e interrogativas. Dentre os arquivos de configuração estão as definições de URIs para elementos importantes da DBPedia, tais como *endpoint*, além das URIs dos principais tipos de dados da ontologia, entre outras.

5.2 FUNCIONAMENTO

Nesta seção, será demonstrado o fluxo de operação da implementação em JavaScript do algoritmo. Conforme mencionado anteriormente, o algoritmo pode iniciar através do recebimento de requisições HTTP POST nas rotas {*endpoint}/sentence/* e {*endpoint}/question/* (Quadro 4). As duas requisições devem enviar um parâmetro do tipo *string* contendo a URI da entidade na ontologia e podem enviar um parâmetro numérico indicando o número de sentenças a serem geradas. Após uma verificação se os parâmetros não são nulos ou contém valores inválidos, a rota aciona a função para criar sentenças da controladora *SentenceController*. Essa função espera receber a URI da entidade, o tipo de sentença a ser gerada (afirmativa ou interrogativa), a quantidade de frases e uma função para chamar após a execução da criação das sentenças.

Quadro 4 – Exemplo de *POST* para o *endpoint* de questões.

```
POST /question HTTP/1.1
Host: localhost:3000
Content-Type: application/x-www-form-urlencoded
quantity=12&entity=http%3A%2F%2Fdbpedia.org%2Fresource%2FUnited_States
```

Fonte: Elaborado pelo autor.

Na função *create* da controladora *SentenceController*, o primeiro passo é verificar se a quantidade de sentenças respeita a configuração de limite máximo de sentenças a serem geradas, pois caso a implementação faça muitas requisições para a DBPedia, a conexão é encerrada após algumas requisições. Caso a quantidade exceda o limite, somente uma sentença será gerada. Ainda na mesma função, a controladora *SentenceController* invoca uma função da controladora *EntityController* para obter as propriedades *rdf:label* e *rdf:type* da entidade. As consultas em SPARQL executadas para obter essas duas informações estão descritas no Quadro 5, consultas *a* e *b*.

Quadro 5 – Consultas SPARQL utilizadas.

a) Exemplo da consulta para obter *rdf:label*
SELECT *?label*
FROM http://dbpedia.org
WHERE {
<http://dbpedia.org/resource/Porto_Alegre> <http://www.w3.org/2000/01/rdf-schema#label> *?label* }
b) Exemplo da consulta para obter *rdf:type*
SELECT *?type*
FROM http://dbpedia.org
WHERE {
<http://dbpedia.org/resource/Porto_Alegre> < http://www.w3.org/1999/02/22-rdf-syntax-ns#type > *?type* }
c) Exemplo da consulta para obter todas as propriedades
SELECT *?predicate, ?object*
FROM http://dbpedia.org
WHERE {
<http://dbpedia.org/resource/Porto_Alegre> *?predicate ?object* }

Fonte: Elaborado pelo autor.

Após obter as informações contidas nas propriedades *rdf:label* e *rdf:type* e preencher corretamente o modelo que representa a entidade (*EntityModel)*, a aplicação segue para o próximo passo que é recuperar todas as relações da entidade (Quadro 5, consulta *c*).

Para cada uma das triplas, o algoritmo prevê critérios para gerar frases em linguagem natural. Logo, somente as triplas com predicado que obteve sucesso nas validações serão enviadas para o passo seguinte. Os critérios são listados no Quadro 6.

Quadro 6 – Critérios para uma tripla com predicado válido para geração de linguagem.

A	O predicado da tripla é uma URI;
B	O predicado da tripla pertence a ontologia primária da aplicação (nesta implementação, a ontologia da DBPedia);
C	O predicado tem que ter definido a propriedade RDF *domain*

Fonte: Elaborado pelo autor.

Para cada predicado que obtiver êxito com os critérios A e B, a implementação irá consultar novamente a DBPedia, através da consulta *c* (Quadro 5), só que dessa vez tendo o predicado como sujeito da consulta. Desta forma, todas as propriedades RDF do predicado da tripla serão retornadas. A partir do resultado da consulta, a implementação instancia um modelo (*RelationModel)* que representa o predicado indicando a relação entre sujeito e objeto. Como atributos deste modelo tem-se o predicado, o objeto, as propriedades *range, type, domain* e *label* da relação. É neste momento que a validação C (Quadro 6) ocorre.

Todas essas validações foram implementadas, pois aumentam consideravelmente o número de sentenças geradas de forma válida. Para encontrar essas restrições, foram realizadas avaliações que serão discutidas no capítulo seguinte.

Por fim, mais uma consulta é realizada na DBPedia antes de iniciar a escolha do *template* linguístico para obter a propriedade *label* do objeto de todas as relações válidas da entidade. Com todas as informações armazenadas no modelo da entidade (*EntityModel),* a aplicação parte para a construção das sentenças.

5.2.1 Substituição das marcações de um *template* linguístico

Primeiramente, a partir do tipo de sentença que deve ser gerado, definido de acordo com a rota da requisição, a aplicação escolhe um *template* da lista de *templates* linguísticos (Apêndice A). Os *templates* estão armazenados na aplicação em um arquivo de utilidades (*templatesUtil*). Eles estão organizados de acordo com tipo de sentença e tipo de dado do objeto. Portanto, existem *templates* específicos para objetos do tipo URI, *string*, numérico e temporal. Com base nessas duas informações, tipo de sentença e tipo de dado do objeto, um *template* é escolhido de forma randômica.

Com o *template* escolhido, a aplicação parte para o processo de substituição das marcações dos *templates* (Quadro 2). As marcações *{S}* e *{ST},* que correspondem respectivamente ao sujeito e o tipo do sujeito, são substituídas de forma simples. Ou seja, as propriedades *label* do sujeito e do tipo do sujeito são colocadas na sentença sem nenhum tratamento. Para o tipo do sujeito, no entanto, como uma instância pode ter mais de um tipo, se ocorrer um cenário que haja mais de um tipo, apenas um é escolhido de forma aleatória.

Para tratamento do apóstrofo (marcação *{AP}*), antes de aplicar o apóstrofo, é verificada a última letra do elemento anterior a ele na sentença. Se for a letra *s,* somente o símbolo "'" é inserido. Caso contrário, o símbolo com a letra *s* é incluído na sentença (*'s*).

Para substituir o objeto, é verificado se o objeto é um URI. Caso afirmativo, a propriedade *label* da instância representada pelo URI é incluída na sentença. Caso o tipo de dado do objeto seja uma data, de forma randômica, um dos seguintes formatos de datas utilizados no Inglês americano é utilizado: "*on MMM DD, YYYY*" ou "*in MMM, YYYY*", onde MMM representa o nome do mês, DD o dia e YYYY o ano. Caso o tipo de dado do objeto seja um literal, o valor do objeto é incluído na sentença.

Para substituir o tipo do objeto (marcação *{OT}*) ocorre o mesmo processo de substituição do tipo de sujeito. Caso haja mais de um tipo para o objeto, apenas um é escolhido de forma randômica.

A marcação *{PRPOSS}* é substituída por um pronome possessivo. Os *templates* que contém pronome possessivo contém regras indicativas de possessão de sujeitos correspondente a terceira pessoa, tanto singular, quanto plural. Deste modo, dependendo do número de objetos que uma tripla possa ter, o pronome é escolhido de acordo com esse número. Para triplas que contém somente um objeto, o pronome possessivo *its* é utilizado para substituição do marcado. Triplas com mais de um objeto, o pronome possessivo *their* é utilizado.

O marcador *{VHA}* está presente em sentenças que contém objetos numéricos. Esse marcador é substituído pelo verbo *to have*. O algoritmo, na substituição deste marcador, flexiona o verbo de acordo a quantidade de sujeitos de uma oração.

A marcação *{TIME}*, em sentenças onde o tipo de dado do objeto é temporal, é substituída por uma duração. No caso, uma subtração entre a datas atual e a data do valor do objeto é calculada e o resultado é incluído na sentença.

Em *templates* para sentenças interrogativas, existe a marcação *{PRON}* que pode ser substituída por um pronome interrogativo dependendo da informação presente na propriedade *range* do predicado. Caso a classe da ontologia referente a uma pessoa seja o valor da propriedade, o pronome "*Who*" é utilizado. Caso seja o valor da propriedade *range* seja temporal, o pronome "*When*" é escolhido. No caso de o valor ser a classe indicativa de lugar dentro da ontologia, o pronome "*Where*" é utilizado. Nos demais casos, o pronome "*What*" é o escolhido.

A marcação *{A}* é indicativa para artigos. Em um *template,* antes de ser substituído pelo valor correto, a aplicação avalia o local que o marcador está e verifica a última letra antes dele. Caso seja uma vogal, o artigo "*an*" é utilizado. Caso contrário, o artigo *"a*".

O dado presente no predicado da tripla pode ser utilizado em dois marcadores: *{P}* e *{PV}*. O marcador *{P}* buscará por sinônimos da propriedade *label* do predicado no WordNet. Essa consulta ocorre através do uso de um módulo do NodeJS que consulta uma cópia local do WordNet. Com a lista de sinônimos retornada, o léxico que contém um indicador de frequência de palavras é consultado para cada sinônimo. A palavra que tem o maior índice de frequência é escolhida para substituição do marcador. Além disso, o algoritmo flexiona a palavra que substituirá o marcador de acordo com o número de objetos que a tripla contém.

Há também uma configuração na implementação do algoritmo que relaciona classes da DBpedia com papéis semânticos no WordNet. Essa configuração tem como objetivo selecionar o *synset* com o mesmo papel semântico da palavra original representada pela relação da tripla.

Para a marcação *{PV}*, novamente uma busca por sinônimos é feita nos mesmos moldes da anterior. Porém, desta vez, para cada um dos itens da lista será efetuada uma consulta no léxico de combinações para encontrar o verbo mais adequado para a palavra. Caso seja encontrado, o verbo passa por uma função para colocá-lo no tempo verbal correto. O tempo verbal das sentenças é o pretérito simples. Logo, uma validação é feita para identificar se o verbo é regular ou irregular. Caso regular, o sufixo "*ed*" é adicionado. Caso contrário, a forma verbal do verbo irregular é utilizada. Essa mesma validação ocorre para substituição do marcador *{VTB}*.

Com todas as substituições realizadas, a sentença é acrescentada na lista que será retornada como resultado da requisição solicitada.

6 AVALIAÇÃO

A pesquisa consistiu em três momentos distintos para a sua elaboração. Em um primeiro momento, foi necessário validar a hipótese de que seria possível gerar linguagem natural a partir do relacionamento entre sujeitos e objetos em uma base modelada com RDF. Em um segundo momento, foi necessário identificar a possibilidade de geração de sentenças apoiadas por um conjunto maior de *templates* linguísticos. Em um terceiro momento foram realizadas melhorias no algoritmo e recursos, tendo como base os resultados obtidos na segunda avaliação. Ou seja, a construção do algoritmo apresentado até o momento é resultado da análise de avaliações realizadas previamente. Os ajustes foram cuidadosamente executados, de acordo com o objetivo deste trabalho. Esses ajustes, por fim, foram então avaliados em uma terceira e última avaliação.

Desta forma, essa seção é dividida em quatro partes. As duas primeiras seções descrevem as avaliações preliminares. Elas serão descritas por serem consideradas importantes na elaboração da pesquisa até este momento, pois elas validaram pontualmente alguns dos diferenciais deste trabalho. A primeira avaliação buscou validar o diferencial de gerar linguagem natural com os dados presentes em bases de dados abertos e conectados, sem apoio de corpus auxiliar, tirando proveito somente das informações embarcadas nas relações entre sujeitos e objetos. Já a segunda avaliação validou parcialmente o segundo diferencial, avaliando o resultado da inclusão de alguns *templates* linguísticos na construção das sentenças.

Logo, a primeira seção discute a primeira avaliação, seus objetivos e resultados encontrados. A segunda seção, por sua vez, discute os mesmos aspectos em relação a segunda avaliação realizada. Já a terceira seção destaca a avaliação final e apresenta de forma mais detalhada o resultado da pesquisa até então. Por fim, a seção final discute a análise dos resultados encontrados até o momento.

6.1 PRIMEIRA AVALIAÇÃO: VALIDAÇÃO DOS CRITÉRIOS PARA SELEÇÃO DE TRIPLAS PARA GERAÇÃO DE FRASES CURTAS

Basicamente a principal preocupação em um primeiro momento foi o processo de seleção de relacionamentos de uma dada entidade descrita com o padrão RDF para geração de sentenças. Ou seja, a partir da entidade que representa “Bill Gates”,

a pergunta feita inicialmente foi: quais são os relacionamentos que têm informações suficientes e relevantes para geração de linguagem?

Para responder esse questionamento, uma primeira versão do algoritmo foi desenvolvida e, consequentemente, esta primeira avaliação foi projetada para validar o resultado de sua execução. Com os resultados obtidos, esperava-se calibrar a lógica de seleção de relacionamentos.

Dito isso, a primeira avaliação foi planejada para gerar até 30 sentenças interrogativas de cinco conceitos. Há mais de 4,5 milhões de instâncias no conjunto de dados de 2014 da DBPedia (DBPEDIA, 2014). Destas, somente cinco classes foram selecionadas de acordo com o seguinte critério: dois conceitos da classe com maior número de instâncias (Pessoas e Lugares), um conceito da classe com o menor número de instâncias (Doenças), e outras duas de forma aleatória (Eventos e Espécies). Este critério foi definido para permitir observar características variadas destes diferentes tipos de entidade com o objetivo de identificar as propriedades de cada tipo que possam vir a se tornar frases curtas em linguagem natural.

Com todas as sentenças geradas, buscou-se identificar como essas sentenças se comportavam em relação a duas dimensões de avaliação. A primeira e a segunda dimensão avaliaram a relevância das informações presentes na sentença e a terceira dimensão é a análise gramatical da sentença gerada. Entende-se por relevância das informações a importância dos dados de uma tripla RDF para determinado público.

A relevância foi importante nesta primeira versão para identificar os critérios de avaliação do algoritmo para escolha de uma tripla para lexicalização, pois algumas triplas não têm informações relevantes para seres humanos. Em outras palavras, existem dentro do conjunto de relacionamentos de uma instância, várias informações trazidas da Wikipedia que, em um primeiro momento, não fazem sentido para leitores humanos. A propriedade da ontologia da DBPedia "*width*", por exemplo, não tem nenhuma importância para o leitor humano, pois não se trata de uma informação relevante e sim da estrutura da linguagem de marcação WEB da Wikipedia.

Além disso, há duas dimensões de relevância: relevância para um público geral e relevância para um público específico. Essa divisão ocorreu em função de algumas instâncias terem informações relevantes para pessoas com um conhecimento específico. Este é o caso, por exemplo, das instâncias da classe "*Disease*". Identificadores CIDs geralmente não tem relevância para o público geral e sim para

profissionais da área da saúde. Portanto, para realizar essa avaliação de relevância, foram convidadas pessoas de diferentes áreas de conhecimento.

Dentro deste contexto, considerando a avaliação gramatical, uma vez que as sentenças seriam avaliadas somente se estavam corretas ou não, não houve necessidade de haver mais de um linguista para avaliação de todas as questões. Para a avaliação da relevância para um público geral, dez pessoas foram convidadas. No entanto, para a avaliação por relevância para um público específico, somente cinco pessoas foram convidadas. O número de avaliadores para a dimensão de relevância foi determinado por conveniência, já que nesta primeira avaliação a ideia era medir a possibilidade de geração de sentenças a partir das informações presentes nas propriedades dos elementos de uma tripla RDF.

Para tanto, foram elaborados três conjuntos de critérios para seleção de triplas para lexicalização. O Quadro 7 descreve os conjuntos e suas características.

Quadro 7 – Conjunto de critérios para seleção de triplas RDF.

ID	**Descrição**
C1	Valida somente se o predicado da tripla contém a propriedade *comment* indicando que o predicado é reservado para a DBPedia
C2	Aplica C1 e valida se a tripla tem seu predicado definido dentro da ontologia da DBPedia
C3	Aplica C2 e valida se o predicado da tripla contém a propriedade *domain* com o valor igual a propriedade *type* do sujeito da tripla.

Fonte: Elaborado pelo autor.

Seguindo essa lógica, um total de 205 sentenças afirmativas foram geradas aplicando somente o conjunto de critérios C1. Executando novamente a implementação, aplicando o conjunto de restrições C2, em uma perspectiva geral, o número de sentenças geradas caiu para mais de 50% comparando com os resultados aplicando C1. Após novamente executar, desta vez aplicando C3, o número de sentenças, em comparação com C1, caiu para mais de 75%. O Quadro 8 apresenta somente alguns exemplos das sentenças geradas e suas avaliações em relação às dimensões estudadas, pois houve algumas dezenas de sentenças nesta primeira avaliação. A lista completa de sentenças desta primeira avaliação está no Apêndice B.

Como pode ser observado no Quadro 8, as sentenças tiveram uma única regra para construção das frases para as triplas selecionadas. O algoritmo se baseou em um *template* de uma frase interrogativa que questiona a possessão do sujeito em relação ao objeto da tripla RDF.

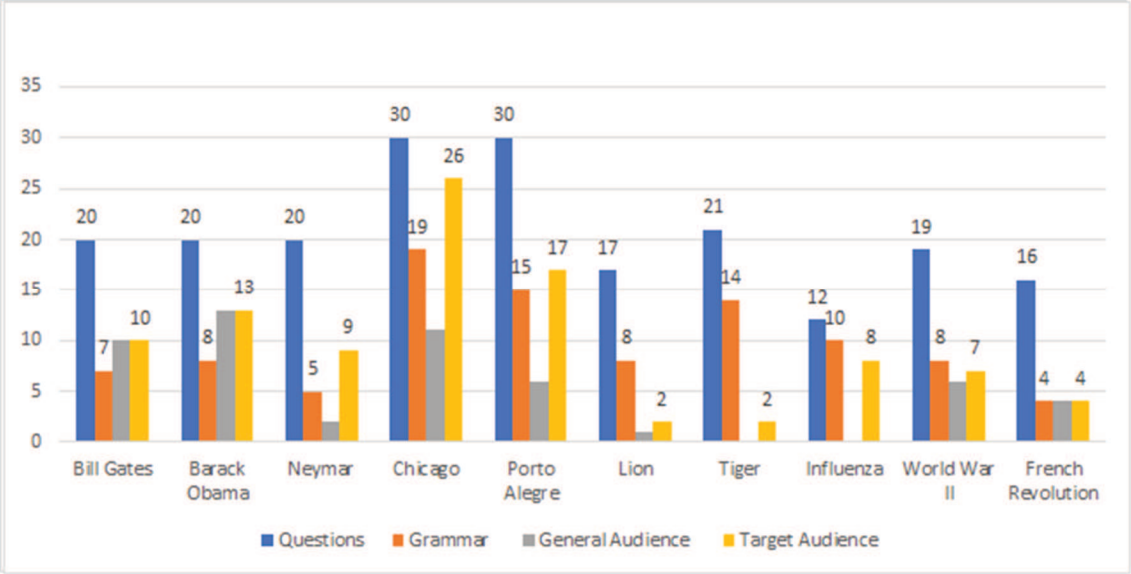
Quadro 8 – Exemplos de sentenças geradas na primeira avaliação.

Sentença	**Gramática**	**Relevância Geral**	**Relevância Público-Alvo**
What's Bill Gates' birth place?	Não	Sim	Sim
What's Barack Obama's alma mater?	Sim	Sim	Sim
What's Neymar's career station?	Sim	Não	Não
What's Chicago's area code?	Sim	Não	Sim
What's Porto Alegre's founding date?	Sim	Não	Sim
What's Lion's status?	Sim	Não	Não
What's Tiger's status?	Sim	Não	Não
What's Influenza's ICD10?	Sim	Não	Sim
What's World War II's combatant?	Não	Sim	Sim
What's French Revolution's date?	Sim	Não	Sim

Fonte: Elaborado pelo autor.

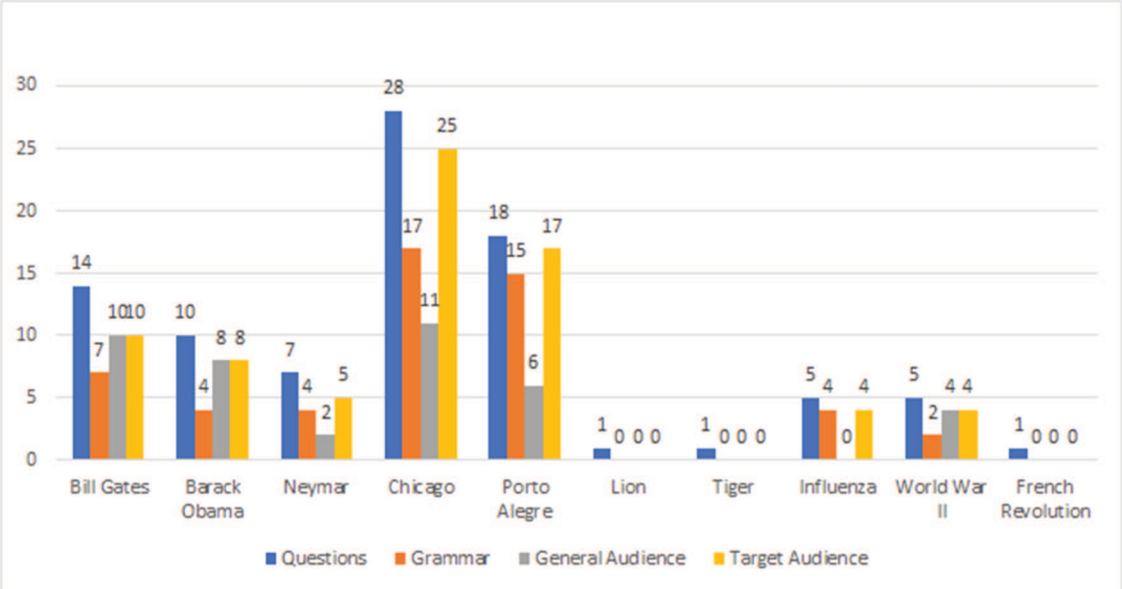
A partir das três listas de sentenças resultantes da aplicação de cada um dos conjuntos de critérios (C1, C2 e C3), as dimensões de gramática, relevância geral e relevância para um público-alvo foram aplicadas em cada lista de sentenças. Nos gráficos a seguir serão exibidas as relações entre o número de questões geradas e as dimensões de avaliação. As séries dos gráficos representam o número de sentenças para cada conjunto de restrições e o número sentenças corretas gramaticalmente, relevantes para um público geral e relevantes para um público específico.

Gráfico 1 – Avaliação das sentenças geradas por C1.



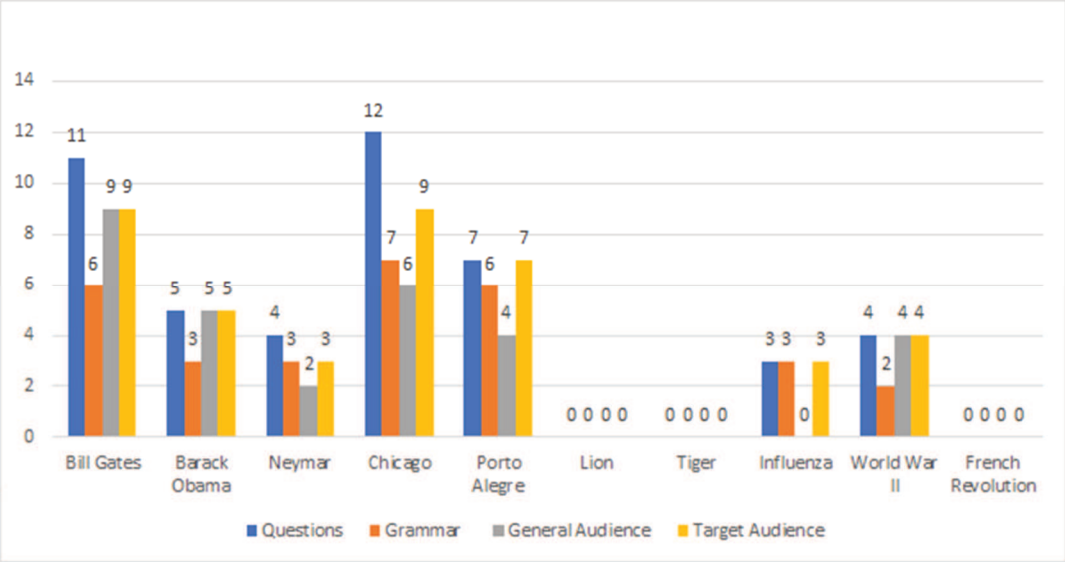
Fonte: Elaborado pelo autor.

Gráfico 2 – Avaliação das sentenças geradas por C2.



Fonte: Elaborado pelo autor.

Gráfico 3 – Avaliação das sentenças geradas por C3.



Fonte: Elaborado pelo autor.

Os Gráficos 1, 2 e 3 mostram que o conjunto C1 gerou mais perguntas incorretas do ponto de vista gramatical e menos relevante para ambos os públicos. Quando avaliado o resultado do conjunto C2, observa-se uma melhora significativa. Entretanto, quando avaliado os resultados do conjunto C3, os resultados são ainda melhores e promissores.

A partir das avaliações observa-se que aplicando um maior número de critérios para seleção das triplas RDF passíveis de lexicalização, há uma tendência que o algoritmo consiga lexicalizar uma tripla, do ponto de vista gramatical, de forma correta, e com informações relevantes para um público geral e um público específico. Ou seja, aplicando restrições que validem as propriedades RDF dos relacionamentos de uma determinada instância dentro de uma ontologia, tais triplas podem ter o conhecimento, por elas representado, transformado em linguagem natural somente com as informações presentes nas propriedades dos elementos da tripla RDF. Além do mais, o resultado desta primeira avaliação indicou a possibilidade de geração de frases afirmativas.

Em suma, conforme já mencionado, esta primeira avaliação forneceu insumos para indicar a possibilidade de geração de linguagem natural a partir das propriedades dos elementos de uma tripla RDF e avaliar as possibilidades para melhoramento do processo de lexicalização.

6.2 SEGUNDA AVALIAÇÃO: INCLUSÃO DE RECURSOS LINGUÍSTICOS

Uma segunda avaliação foi realizada para validar uma série de modificações no algoritmo a partir do resultado da primeira avaliação e a inclusão de alguns recursos linguísticos.

Desta forma, um novo método de avaliação foi planejado. Para esta avaliação, o algoritmo foi ampliado para gerar sentenças afirmativas e interrogativas, o que demandou que os métodos de avaliação fossem adaptados. Somente um linguista foi convidado para avaliar as sentenças. Entretanto, um grupo de estudantes de graduação falantes nativos do idioma inglês também avaliou os resultados. A DBpedia foi novamente utilizada como repositório de dados abertos e conectados.

Esta segunda avaliação gerou um total de 84 sentenças afirmativas e 84 sentenças interrogativas. No Quadro 9 é possível visualizar algumas das sentenças geradas. O conjunto total de sentenças estão nos Apêndices C e D.

Quadro 9 – Exemplos sentenças geradas na segunda avaliação.

Sentença
Barack Obama is a person whose birth place is Hawaii.
Bill Gates' attended Harvard University.
Neymar's birth place is São Paulo.
Elvis Presley's death year was 1977.
Arnold Schwarzenegger's went to University of Wisconsin–Superior.
What is the birth place of Charlie Chaplin?
What is the name of Brazil's capital?
What is the Seattle's commander name?
What is Porto Alegre's UTC offset?
What is United States' official language?

Fonte: Elaborado pelo autor.

Para geração destas sentenças, até 15 relacionamentos foram escolhidos de conceitos das classes Pessoas e Lugares. Um total de seis conceitos de pessoas foi selecionado enquanto quatro conceitos de lugares foram selecionados. Estes conceitos estão descritos no Quadro 10. Nele também pode ser observada a classe,

o conceito e o URI que permite o acesso ao material utilizado, tal como está armazenado na DBpedia.

Quadro 10 – Entidades utilizadas na segunda avaliação para geração de frases curtas.

Classe	**Conceito**	**URI**
Pessoa	Barack Obama	*http://dbpedia.org/page/Barack_Obama*
Pessoa	Bill Gates	*http://dbpedia.org/page/Bill_Gates*
Pessoa	Neymar	*http://dbpedia.org/page/Neymar*
Pessoa	Elvis Presley	*http://dbpedia.org/page/Elvis_Presley*
Pessoa	Arnold Schwarzenegger	*http://dbpedia.org/page/Arnold_Schwarzenegger*
Pessoa	Charles Chaplin	*http://dbpedia.org/page/Charles_Chaplin*
Lugar	Brazil	*http://dbpedia.org/page/Brazil*
Lugar	Seattle	*http://dbpedia.org/page/Seattle*
Lugar	Porto Alegre	*http://dbpedia.org/page/Porto_Alegre*
Lugar	United States	*http://dbpedia.org/page/United_States*

Fonte: Elaborado pelo autor.

Para a obtenção de resultados, dado que o fator de subjetividade pode ter um impacto significativo, foram elaborados e apresentados aos avaliadores critérios balizadores. Todos os participantes da avaliação levaram em consideração cinco critérios para suas avaliações.

Os critérios utilizados são os seguintes:

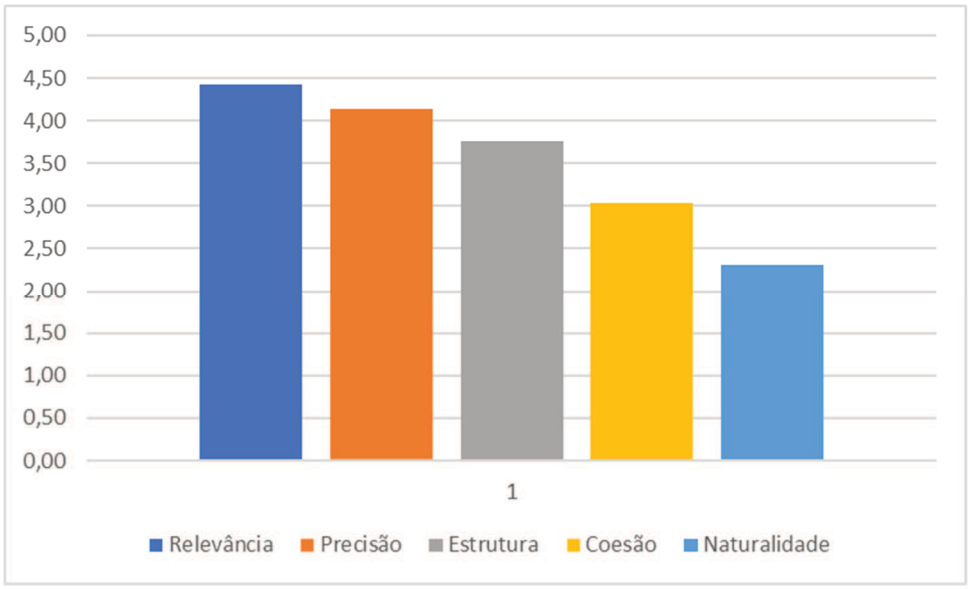
- Relevância: Quão relevante a informação na sentença é para você?
- Precisão: Quão precisas as informações na sentença são para você?
- Estrutura: A sentença está gerada corretamente em termos de gramática?
- Coesão: A sentença está estruturada de uma forma lógica?
- Naturalidade: A sentença soa como uma frase que você ouviria no seu dia-a-dia?

Cada participante recebeu uma planilha contendo todas as sentenças geradas pelo algoritmo com a tripla que originou a sentença. Segundo os critérios acima, o

participante tinha que qualificar cada sentença entre os valores de 1 a 5, sendo 1 a menor nota e 5 a maior nota.

Dentro deste contexto, considera-se que uma sentença foi gerada de forma apropriada se obteve uma média das notas das dimensões maior que 3. Observa-se no Gráfico 4, que a dimensão que teve a média das notas de todas as sentenças mais baixa foi a naturalidade. Mesmo assim, os participantes da avaliação concluíram que aproximadamente 66% das sentenças geradas foram consideradas apropriadas em relação aos dados obtidos a partir das triplas.

Gráfico 4 – Média das notas da avaliação das sentenças da segunda avaliação.



Fonte: Elaborado pelo autor.

Dentre os outros 44% das sentenças que não foram consideradas apropriadas, foram percebidos erros em função do uso do léxico de sinônimos. Por exemplo, para uma determinada sentença, ao buscar um sinônimo para a propriedade *label* do relacionamento *children* de um dos conceitos, o resultado foi a palavra *cub*, que tem um significado diferente da informação contida na tripla. Porém, de acordo com os avaliadores, a sentença em questão foi construída de forma correta pelo algoritmo, faltando somente coerência na escolha do sinônimo. O caso em questão foi a sentença "*Elvis Presley's cub is Lisa Marie Presley*".

Outra observação importante nos 44% das sentenças que foram consideradas inapropriadas, foi o uso de palavras pouco usuais para a relação *alma mater* de alguns conceitos. Ao consultar o léxico, o algoritmo escolheu palavras que não soaram

naturais em uma conversa do dia-a-dia. Por exemplo, "*Barack Obama's institution is Occidental College*".

O algoritmo criou também sentenças utilizando um *template* contendo o pronome *whose* de forma correta do ponto de vista gramatical, mas que não soavam naturais para os avaliadores. Exemplos destes casos são as sentenças: "*Bill Gates is a person whose residence is Medina*" e "*Barack Obama is a person whose birth place is Hawai*".

No geral, mais da metade das sentenças (66%) foi bem avaliada no ponto de vista gramatical pelos avaliadores, mas uma grande porção teve dificuldades para alcançar a naturalidade. Ou seja, os resultados foram promissores fornecendo indicações para melhorias com o intuito de aumentar o número de sentenças consideradas apropriadas.

Em suma, esta segunda avaliação validou a primeira tentativa de inclusão dos recursos linguísticos no fluxo do algoritmo. A avaliação indicou uma série de melhorias que puderam ser analisadas e implementadas posteriormente.

6.3 TERCEIRA AVALIAÇÃO: ANÁLISE FINAL

A terceira avaliação foi realizada com o algoritmo descrito no capítulo 4, já tendo sido incluídos todos os elementos de melhoria observados com as avaliações anteriores. Para esta última avaliação, um novo modelo de avaliação foi projetado. Um grupo de linguistas foi convidado para participar de uma nova avaliação que foi conduzida durante dois meses. Eles avaliaram um conjunto de 76 sentenças afirmativas e 76 sentenças interrogativas. Nas próximas seções, serão descritos os detalhes da avaliação.

6.3.1 Avaliadores

A primeira pergunta que surgiu na elaboração desta última avaliação foi escolher entre avaliadores humanos ou métricas automatizadas para avaliar o conjunto de sentenças geradas pelo algoritmo. Neste sentido, optou-se por avaliar as sentenças com pessoas ao invés de utilizar métricas automatizadas, pois, atualmente, as métricas automatizadas presentes na literatura, tais como BLEU (PAPINENI, 2002) ou ROUGE (LIN, 2004), não refletem suficientemente a avaliação humana

(NOVIKOVA et al., 2017). As métricas automatizadas comumente utilizadas têm correlação baixa com o julgamento de seres humanos (LIU et al., 2016). Dado esse contexto, somado ao fato que esperava-se avaliar a naturalidade das sentenças geradas, que é uma avaliação subjetiva, optou-se pela avaliação com pessoas.

Dentro deste contexto, profissionais da área da linguística foram cuidadosamente escolhidos e convidados para avaliar as sentenças. Todos os avaliadores são formados e atuam/atuaram na área. Além disso, dentre os avaliadores, foram convidados linguistas nativos americanos para avaliar as sentenças.

Desta forma, para avaliação desta última versão do algoritmo, foram convidados um total de 10 linguistas. Esse número foi definido por conveniência. Destes, 6 são falantes nativos do idioma inglês, enquanto os outros 4 avaliadores são brasileiros. Todos os avaliadores são ou eram professores de Inglês. 80% dos avaliadores estão cursando ou já têm pós-graduação. O Quadro 11 descreve maiores detalhes dos avaliadores.

Quadro 11 – Relação de avaliadores.

Avaliador	**Conceito**	**Profissão**	**Residência**
A1	Mestre	Professor de Inglês	Fairfax, VA - EUA
A2	Mestre	Professor aposentado de escola de idiomas	Fairfax, VA - EUA
A3	Mestre	Professor aposentado de escola de idiomas	Petaluma, CA - EUA
A4	Mestre	Fazendeiro	San Jose, Costa Rica
A5	Mestre	Professor de Inglês	Fairfax, VA - EUA
A6	Mestre	Professor de Inglês	Fairfax, VA - EUA
A7	Bacharel	Professor de Inglês	Charqueadas, RS - Brasil
A8	Doutorando	Psicólogo	Faro, Portugal
A9	Mestrando	Professor de Inglês	São Leopoldo, RS - Brasil
A10	Mestre	Professor de Inglês	RS, Brasil

Fonte: Elaborado pelo autor.

6.3.2 Modelo de avaliação

Todos os avaliadores receberam uma planilha por e-mail contendo as 152 sentenças para avaliação. Diferentemente das avaliações anteriores, essa última mediu somente duas dimensões para cada sentença, pois as dimensões mensuradas na segunda avaliação, por exemplo, avaliavam aspectos fora do escopo do projeto do algoritmo. Um exemplo mais claro é a acurácia da informação. O algoritmo não faz nenhum tratamento em relação a acurácia das informações que são extraídas da BDAC por considerar que isto está fora do escopo do algoritmo.

Os avaliadores precisavam determinar o nível gramatical e de naturalidade de cada sentença com uma nota de 1 a 5, sendo a nota 1 indicativa de uma sentença muito ruim e 5 uma sentença muito boa.

Os avaliadores tiverem 60 dias para avaliar o material e foram solicitados a retorná-lo por e-mail após a conclusão da avaliação. O período de 60 dias foi acordado com todos em função de suas atividades. Alguns são professores e estavam no período de avaliação do meio do semestre em suas universidades. Além disso, foi aberto um método de comunicação para responder dúvidas e questionamentos de todos os avaliadores. Neste canal de comunicação, surgiram algumas dúvidas. Algumas relacionadas ao funcionamento da avaliação e outras mais técnicas. As técnicas foram questionamentos sobre acurácia e formatação dos dados recuperados da BDAC, pois foram criaram sentenças pouco usuais em alguns casos.

6.3.3 Método de análise

Conforme já mencionado, foram geradas 152 sentenças (Apêndice E e F). Metade delas sendo afirmativas e, a outra metade, interrogativas. Além disso, as sentenças foram divididas em quatro grupos para facilitar a análise dos resultados. Os critérios para criação de grupos e seleção para pertencer a eles foram a classe da instância para quais as sentenças foram geradas e o tipo de sentença. Deste modo, criou-se quatro grupos: sentenças afirmativas para instâncias da classe pessoa (SAP); sentenças afirmativas para instâncias da classe lugar (SAL); sentenças interrogativas para instâncias da classe pessoa (SIP); e, por fim, sentenças interrogativas para instâncias da classe lugar (SIL).

Uma vez que cada sentença teve um grau de naturalidade e gramática definido por cada avaliador, um dos objetos de análise foi a média das notas para cada sentença. Também foram aplicados outros recursos estatísticos para ajudar na interpretação dos resultados. Neste sentido, foram calculadas a mediana e o desvio-padrão das médias das notas atribuídas pelos avaliadores para cada sentença. Desta forma, foi possível entender a variância entre as notas das sentenças.

Além disso, foi utilizado um método para medir o grau de concordância entre os avaliadores que avaliam um mesmo conjunto de objetos em uma escala nominal. Normalmente, para utilização de medição do grau de concordância, o método de estatístico de Kappa é utilizado. Esse método tem diversas apresentações. O primeiro foi proposto por Cohen, em 1960. Eventualmente, esta forma de medição foi estruturada para incorporar pesos em suas avaliações (Cohen, 1968).

Porém, ambas comportam, em suas fórmulas matemáticas, somente dois avaliadores. Para permitir que um conjunto de itens pudessem ser avaliados por mais de dois avaliadores, Fleiss, em 1971, propôs um novo método de Kappa. Ele desenvolveu um novo método de estatística de Kappa que generaliza o método proposto por Cohen em 1960. Essa nova formulação permite avaliar a concordância que N avaliadores possam ter para n sujeitos avaliados em K categorias. O teste de Kappa tem colecionado algumas críticas em relação a suas interpretações. Landis e Kock em 1977 montaram uma tabela para interpretação dos resultados sem muitas evidências que as suportasse. O importante, pelo cálculo, é que quanto mais próximo do número um o coeficiente de Kappa estiver, maior será a concordância entre os avaliadores. Se o coeficiente for abaixo de 0, o cenário é total discordância entre os avaliadores. É importante também destacar que, pelas formulações apresentadas por Fleiss, quanto maior a quantidade de categorias, maior a tendência de discordância. Outro ponto importante a destacar é que as categorias, neste modelo de avaliação, por mais que sejam representadas com valores inteiros, elas representam categorias nominais: muito ruim (1), ruim (2), ok (3), bom (4), e muito bom (5).

Para analisar os resultados desta nova etapa de avaliação do algoritmo, as avaliações foram divididas em dois grupos: falantes nativos e falantes não-nativos. Elas foram divididas em função de avaliar a possibilidade de percepção de naturalidade entre avaliadores nativos e não nativos.

Desta forma, as métricas utilizadas foram aplicadas em três conjuntos de respostas: falantes nativos, falantes não-nativos e geral. O conjunto de resultados

geral contém todas as avaliações. Sem diferenciar entre falantes nativos e não-nativos. Desta forma, o conjunto de respostas CR1 contém as respostas das avaliações de seis avaliadores americanos. O conjunto de respostas CR2 contém as respostas das avaliações de quatro avaliadores brasileiros. E, por fim, o conjunto de respostas CR3 contém as dez avaliações.

Em suma, são duas dimensões de avaliação (gramática e naturalidade) para os quatro conjuntos de sentenças. Logo, são 8 conjuntos de valores estatísticos para cada conjunto de respostas. O Quadro 12 ajuda na identificação do conjunto de resultados avaliados para cada um dos conjuntos de respostas.

Quadro 12 – Conjunto de valores estatísticos para cada conjunto de respostas.

Conjunto de Sentenças	**Dimensão de avaliação**	**Métricas**
SAP	Gramática	Para cada linha: Coef. Kappa Fleiss (1973) Média Mediana Desvio-Padrão
SAP	Naturalidade	
SAL	Gramática	
SAL	Naturalidade	
SIP	Gramática	
SIP	Naturalidade	
SIL	Gramática	
SIL	Naturalidade	

Fonte: Elaborado pelo autor.

6.3.4 Apresentação dos resultados

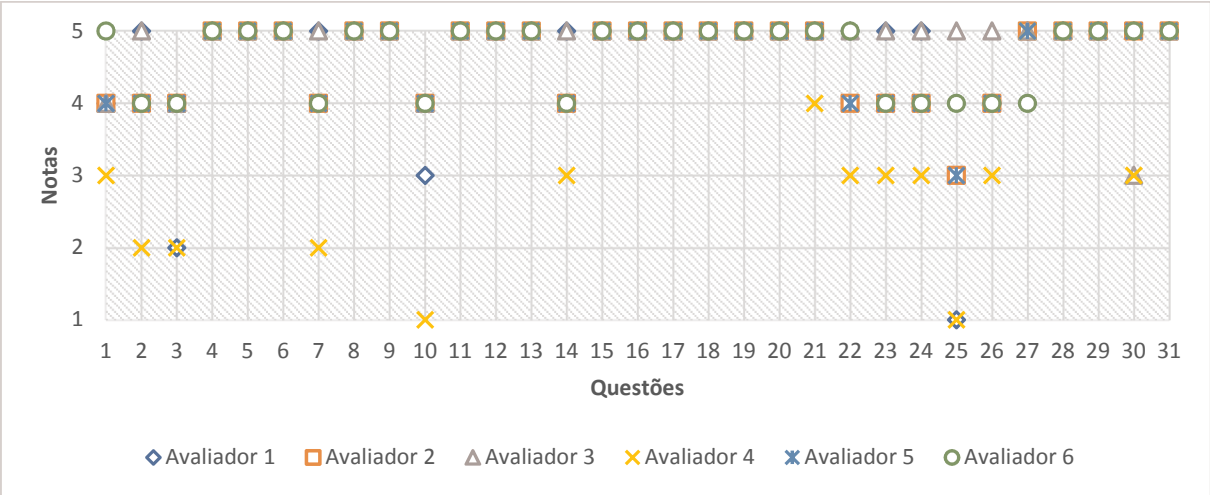
Os itens abaixo, de 6.3.4.1 até 6.3.4.3 apresentam os resultados detalhados para os itens de gramática e naturalidade, para cada um dos 4 conjuntos de sentenças.

6.3.4.1 Falantes nativos (CR1)

Os Gráficos de 5 a 12 representam as avaliações dos falantes nativos para as dimensões gramatical e natural das sentenças geradas pelo algoritmo. A Tabela 1,

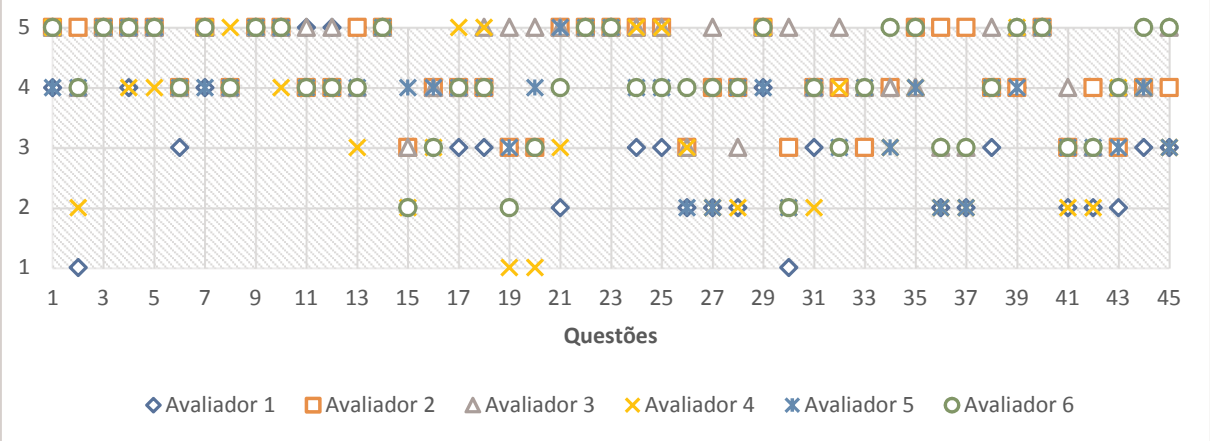
por sua vez, concentra o resultado da aplicação das métricas de avaliação para este grupo de avaliadores.

Gráfico 5 – Avaliação da gramática do conjunto de Sentenças SAP.



Fonte: Elaborado pelo autor.

Gráfico 6 – Avaliação da gramática do conjunto de Sentenças SAL.



Fonte: Elaborado pelo autor.

Gráfico 7 – Avaliação da gramática do conjunto de Sentenças SIP.

Fonte: Elaborado pelo autor.

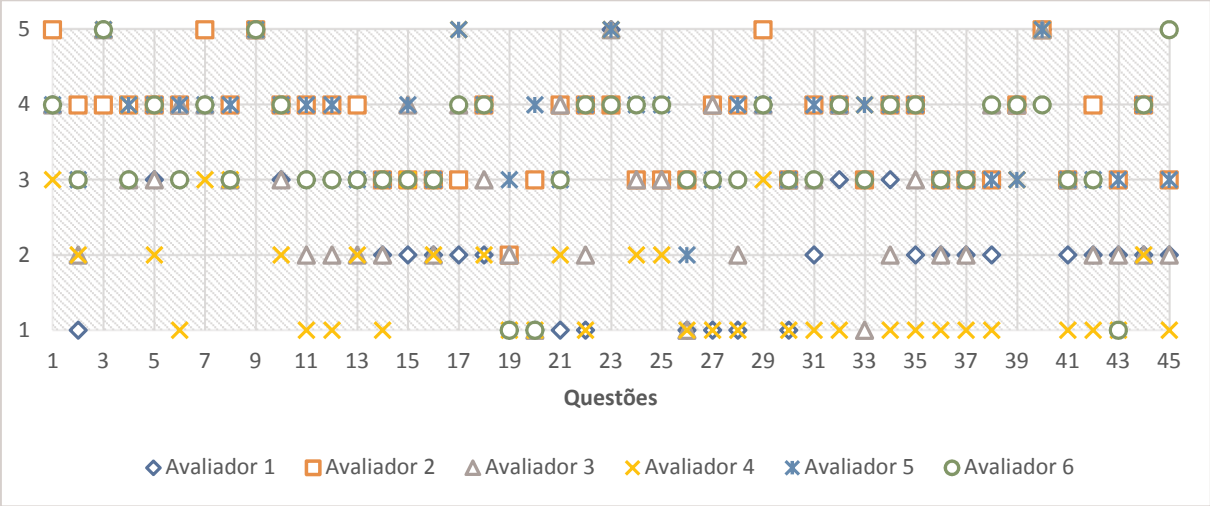
Gráfico 8 – Avaliação da gramática do conjunto de Sentenças SIL.

Fonte: Elaborado pelo autor.

Gráfico 9 – Avaliação da naturalidade do conjunto de Sentenças SAP.

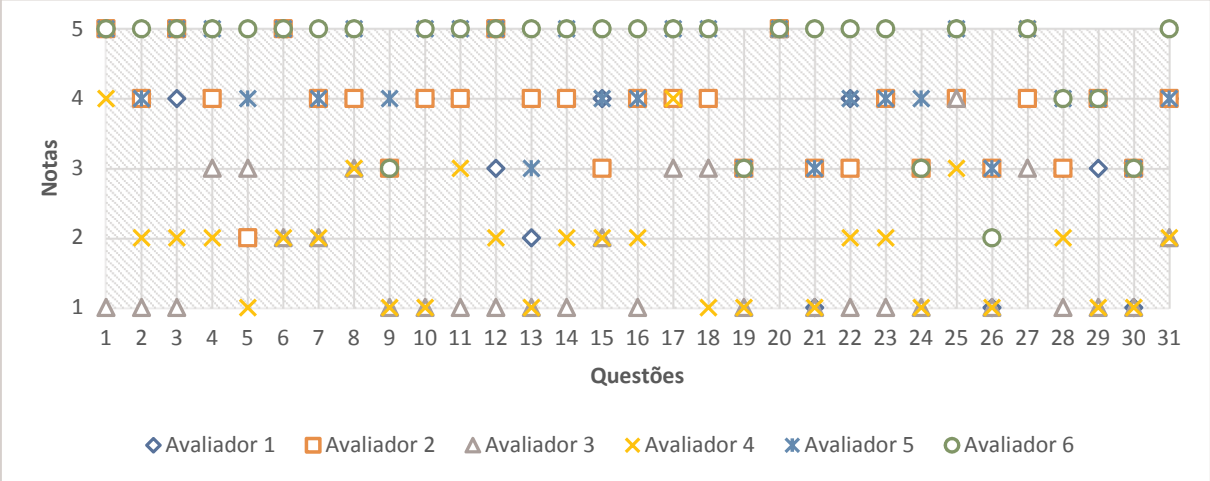
Fonte: Elaborado pelo autor.

Gráfico 10 – Avaliação da naturalidade do conjunto de Sentenças SAL.



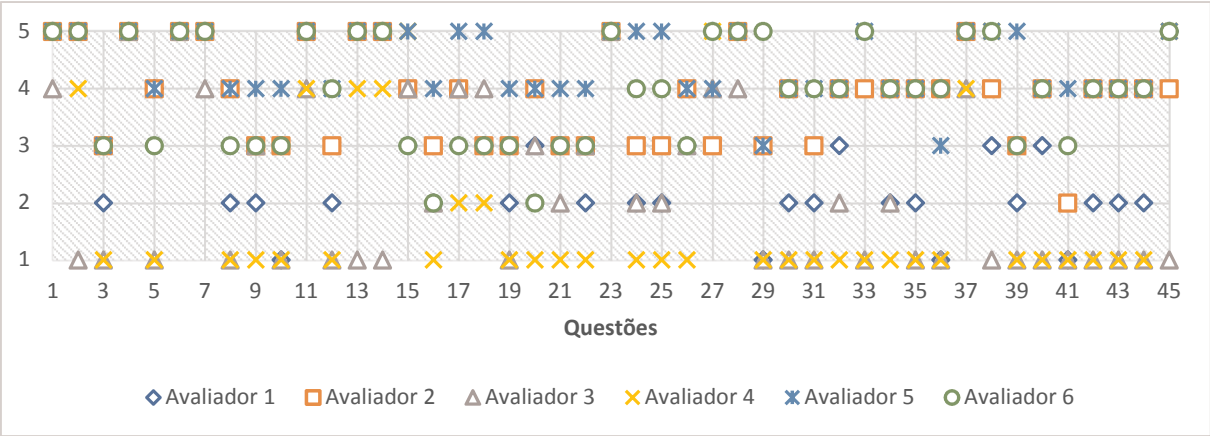
Fonte: Elaborado pelo autor.

Gráfico 11 – Avaliação da naturalidade do conjunto de Sentenças SIP.



Fonte: Elaborado pelo autor.

Gráfico 12 – Avaliação da naturalidade do conjunto de Sentenças SIL.



Fonte: Elaborado pelo autor.

Tabela 1 – Valores das métricas da avaliação da CR1.

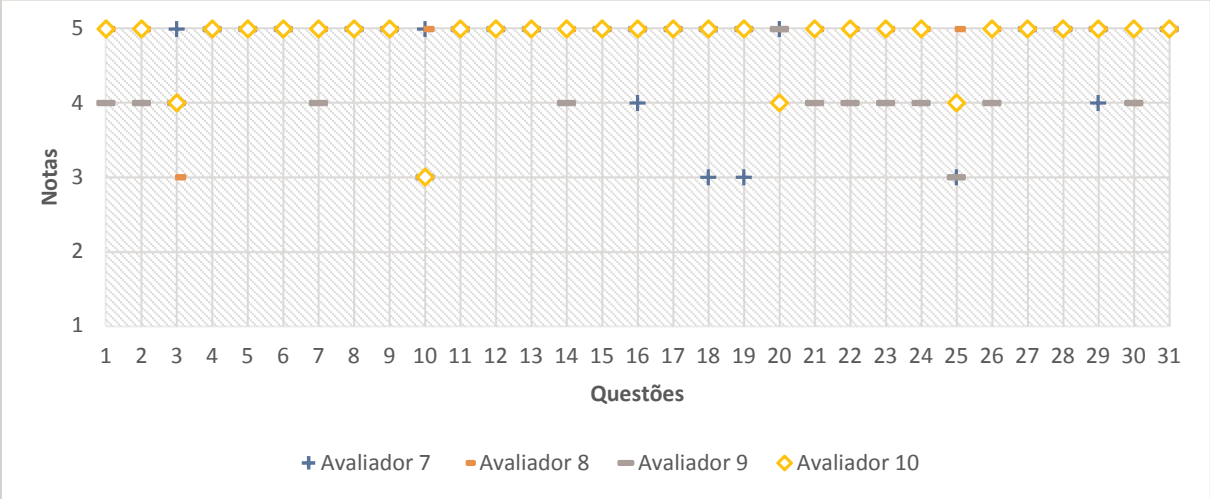
Dimensão de avaliação	Conjunto de Sentenças	Métricas estatísticas	Valores
Gramática	SAP	Média das notas	4,560
		Mediana	5
		Desvio-padrão	0,602
		Concordância	0,376
	SAL	Média das notas	3,940
		Mediana	4
		Desvio-padrão	0,760
		Concordância	0,217
	SIP	Média das notas	4,521
		Mediana	4,667
		Desvio-padrão	0,628
		Concordância	0,155
	SIL	Média das notas	3,863
		Mediana	3,667
		Desvio-padrão	0,814
		Concordância	0,274
Naturalidade	SAP	Média das notas	3,650
		Mediana	3,667
		Desvio-padrão	0,762
		Concordância	0,045
	SAL	Média das notas	3,059
		Mediana	3
		Desvio-padrão	0,779
		Concordância	0,094
	SIP	Média das notas	3,344
		Mediana	3,3333
		Desvio-padrão	0,738
		Concordância	0,046
	SIL	Média das notas	3,289
		Mediana	2,8
		Desvio-padrão	0,938
		Concordância	0,130

Fonte: Elaborado pelo autor.

6.3.4.2 Falantes não-nativos (CR2)

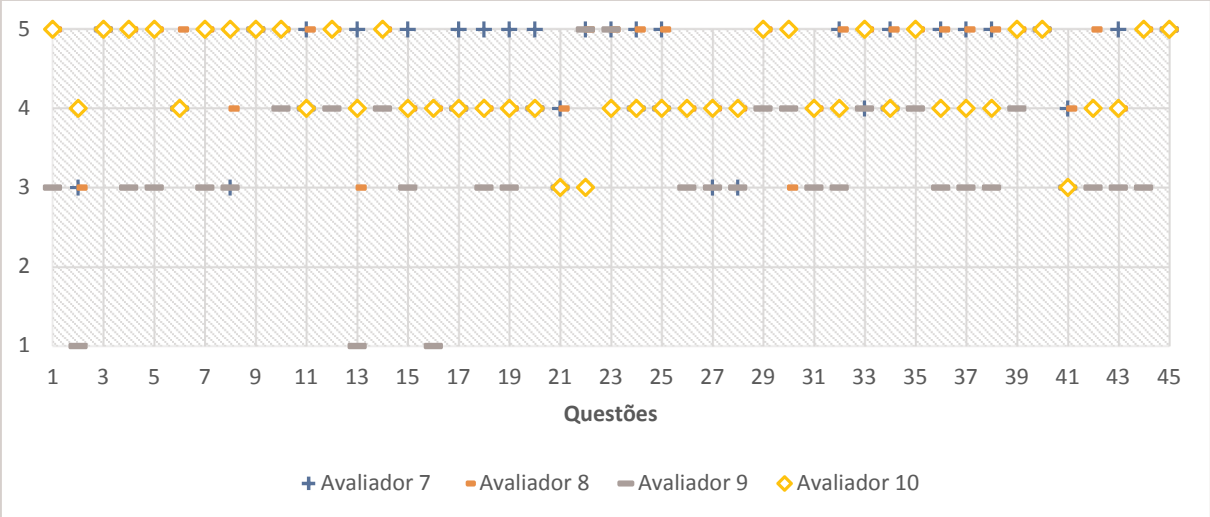
Os Gráficos de 13 a 20 representam as avaliações dos falantes não-nativos para as dimensões gramatical e natural das sentenças geradas pelo algoritmo. A Tabela 2, por sua vez, concentra o resultado da aplicação das métricas de avaliação para este grupo de avaliadores.

Gráfico 13 – Avaliação da gramática do conjunto de Sentenças SAP.



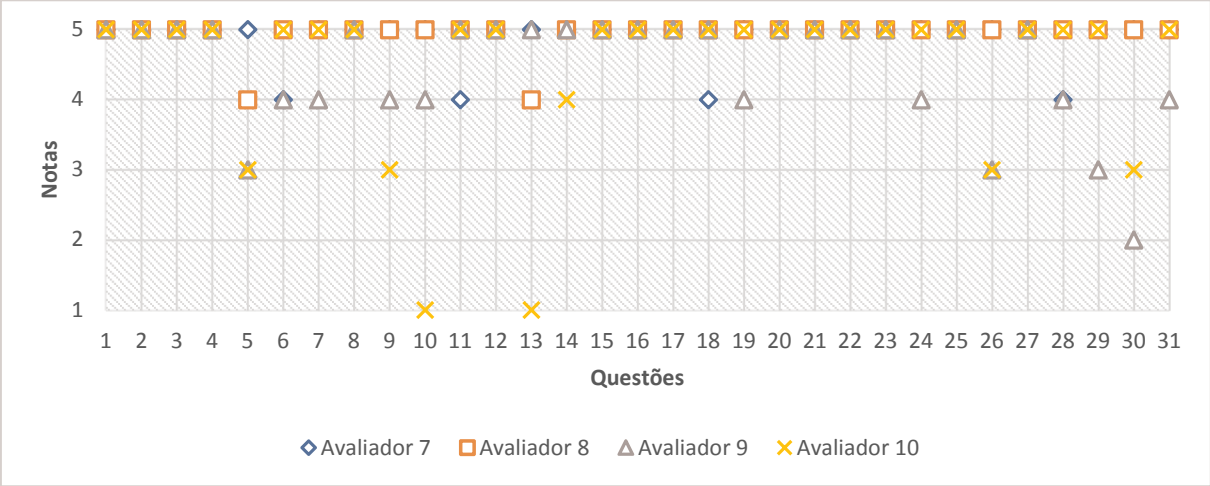
Fonte: Elaborado pelo autor.

Gráfico 14 – Avaliação da gramática do conjunto de Sentenças SAL.



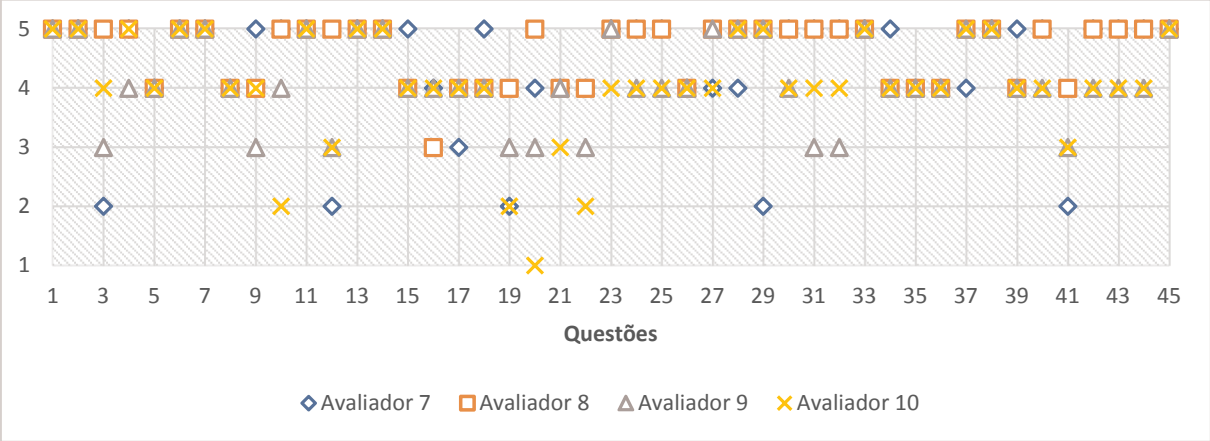
Fonte: Elaborado pelo autor.

Gráfico 15 – Avaliação da gramática do conjunto de Sentenças SIP.



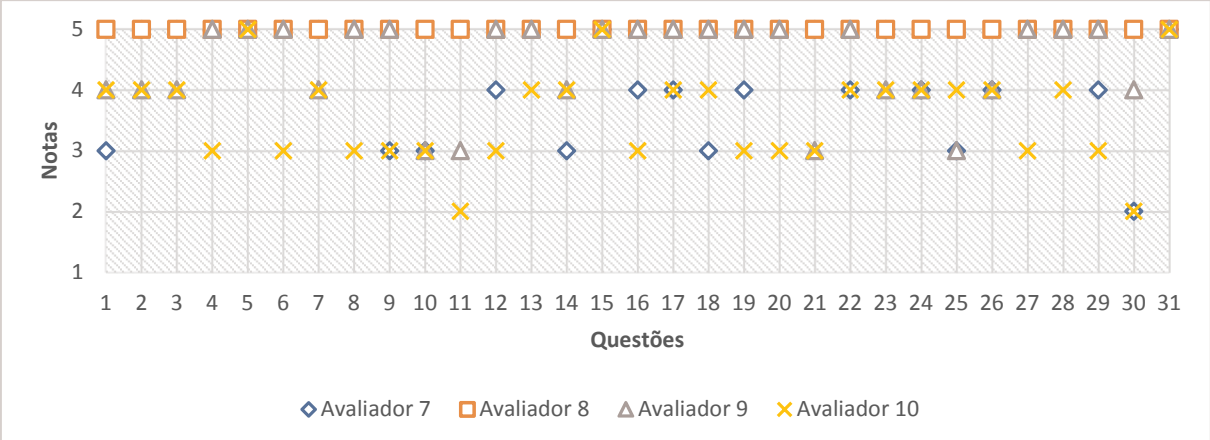
Fonte: Elaborado pelo autor.

Gráfico 16 – Avaliação da gramática do conjunto de Sentenças SIL.



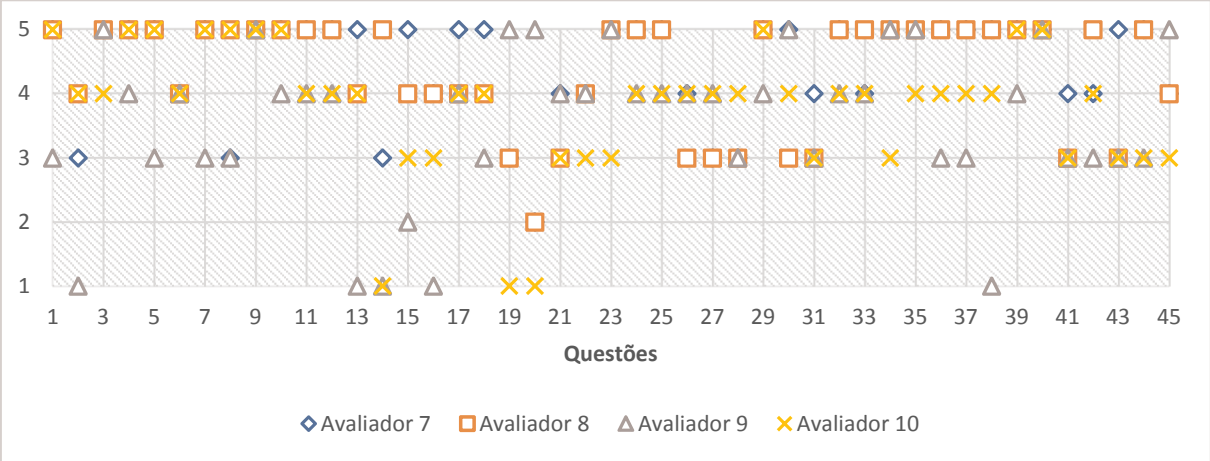
Fonte: Elaborado pelo autor.

Gráfico 17 – Avaliação da naturalidade do conjunto de Sentenças SAP.



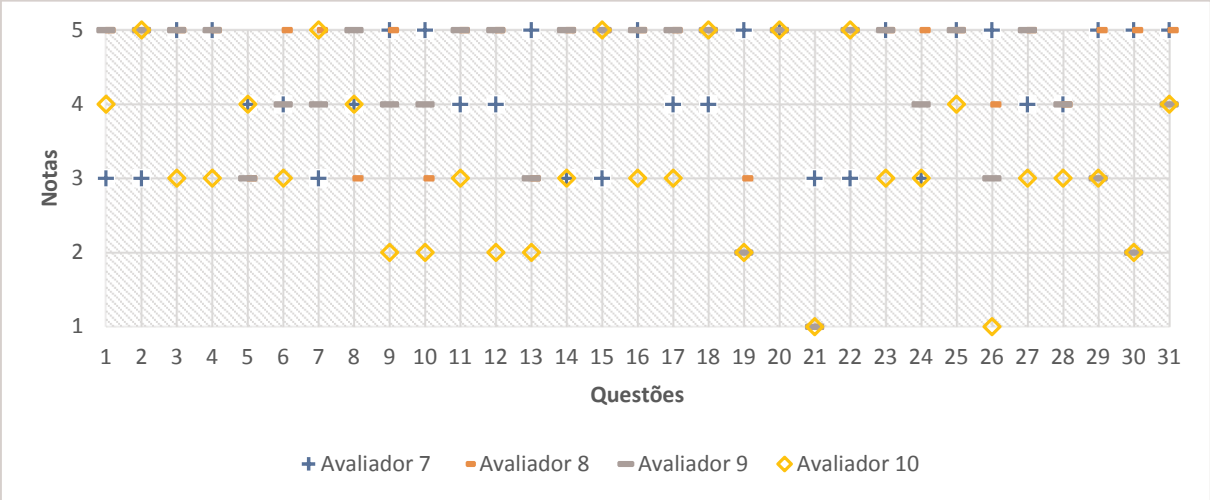
Fonte: Elaborado pelo autor.

Gráfico 18 – Avaliação da naturalidade do conjunto de Sentenças SAL.



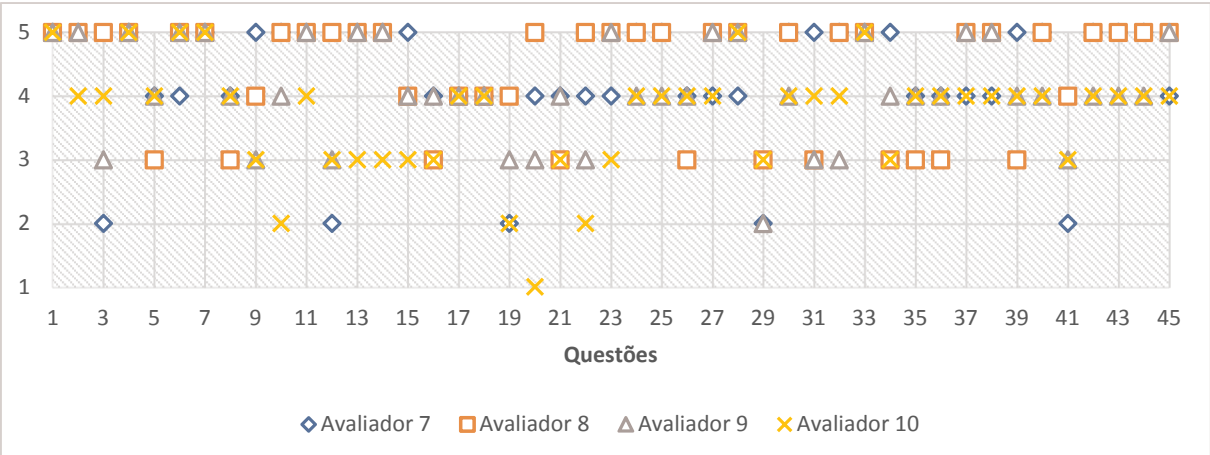
Fonte: Elaborado pelo autor.

Gráfico 19 – Avaliação da naturalidade do conjunto de Sentenças SIP.



Fonte: Elaborado pelo autor.

Gráfico 20 – Avaliação da naturalidade do conjunto de Sentenças SIL.



Fonte: Elaborado pelo autor.

Tabela 2 – Valores das métricas da avaliação da CR2.

Dimensão de avaliação	Conjunto de Sentenças	Métricas estatísticas	Valores encontrados
Gramática	SAP	Média das notas	4,750
		Mediana	4,75
		Desvio-padrão	0,318
		Concordância	0,015
	SAL	Média das notas	4,250
		Mediana	4,25
		Desvio-padrão	0,522
		Concordância	0,069
	SIP	Média das notas	4,677
		Mediana	4,75
		Desvio-padrão	0,432
		Concordância	0,048
	SIL	Média das notas	4,289
		Mediana	4,25
		Desvio-padrão	0,589
		Concordância	0,270
Naturalidade	SAP	Média das notas	4,323
		Mediana	4,5
		Desvio-padrão	0,403
		Concordância	0,004
	SAL	Média das notas	4,028
		Mediana	4,25
		Desvio-padrão	0,646
		Concordância	0,113
	SIP	Média das notas	4,024
		Mediana	4,25
		Desvio-padrão	0,664
		Concordância	-0,001
	SIL	Média das notas	4,106
		Mediana	4,3
		Desvio-padrão	0,607
		Concordância	0,104

Fonte: Elaborado pelo autor.

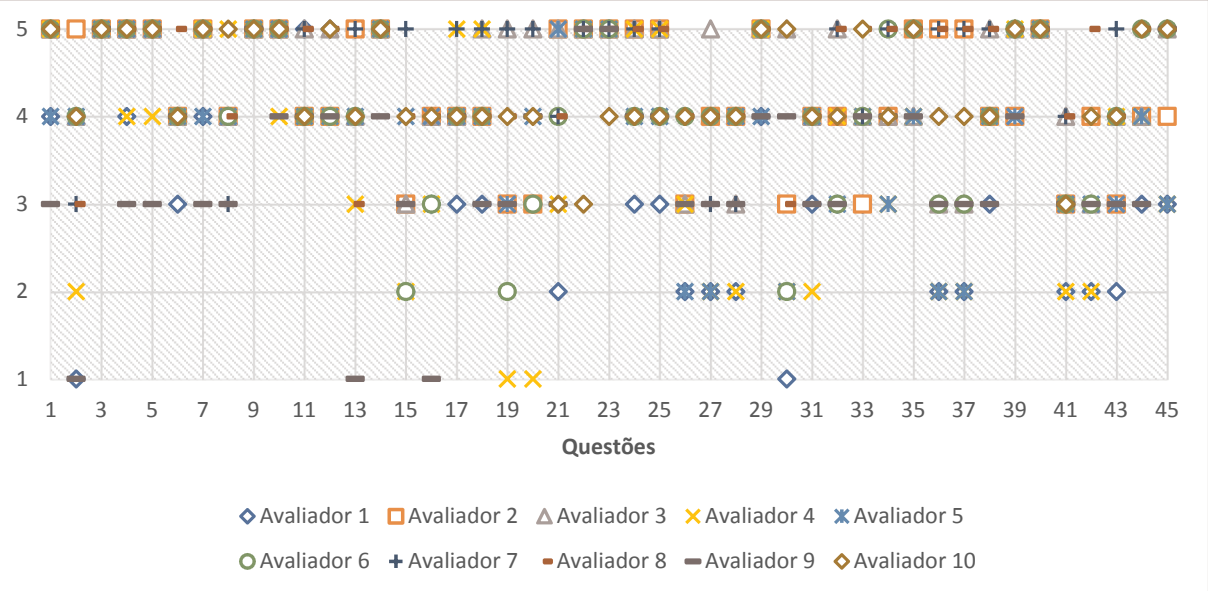
6.3.4.3 Todos os participantes (CR3)

Os Gráficos de 21 a 28 representam as avaliações de todos os participantes para as dimensões gramatical e natural das sentenças geradas pelo algoritmo. A Tabela 3, por sua vez, concentra o resultado da aplicação das métricas de avaliação para a totalidade das sentenças avaliadas por todos os participantes.

Gráfico 21 – Avaliação da gramática do conjunto de Sentenças SAP.

Fonte: Elaborado pelo autor.

Gráfico 22 – Avaliação da gramática do conjunto de Sentenças SAL.

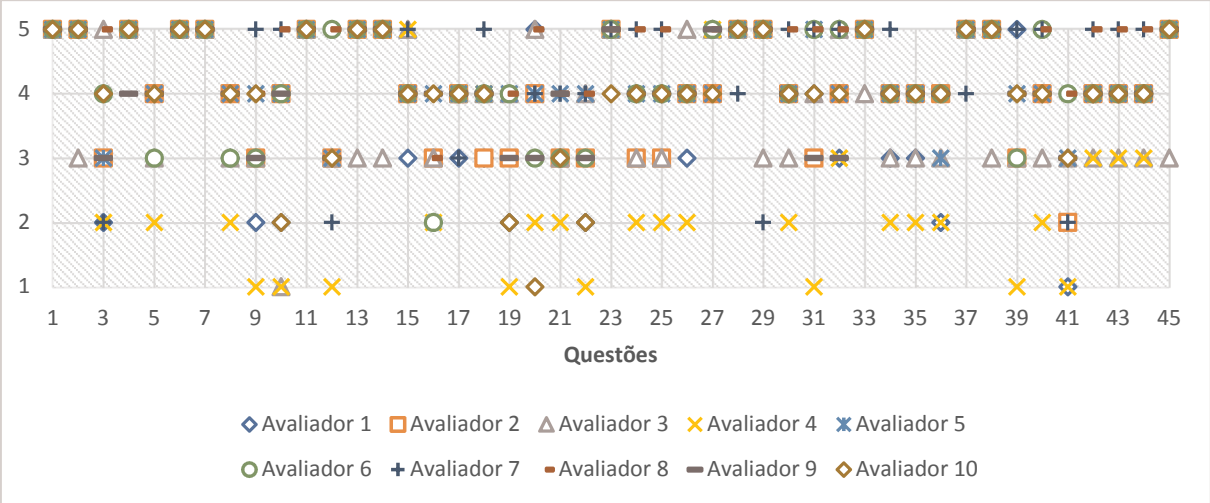


Fonte: Elaborado pelo autor.

Gráfico 23 – Avaliação da gramática do conjunto de Sentenças SIP.

Fonte: Elaborado pelo autor.

Gráfico 24 – Avaliação da gramática do conjunto de Sentenças SIL.

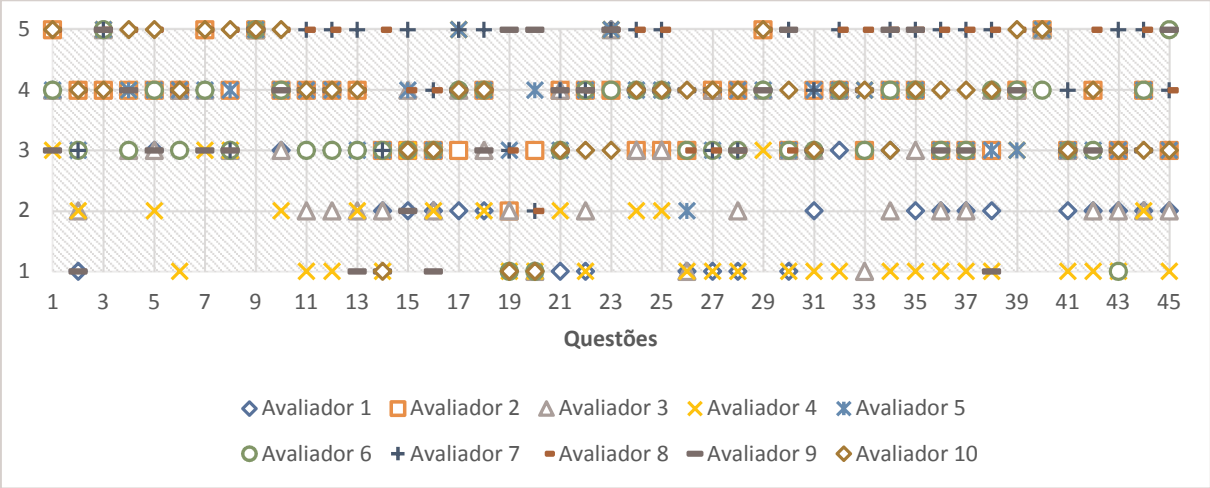


Fonte: Elaborado pelo autor.

Gráfico 25 – Avaliação da naturalidade do conjunto de Sentenças SAP.

Fonte: Elaborado pelo autor.

Gráfico 26 – Avaliação da naturalidade do conjunto de Sentenças SAL.

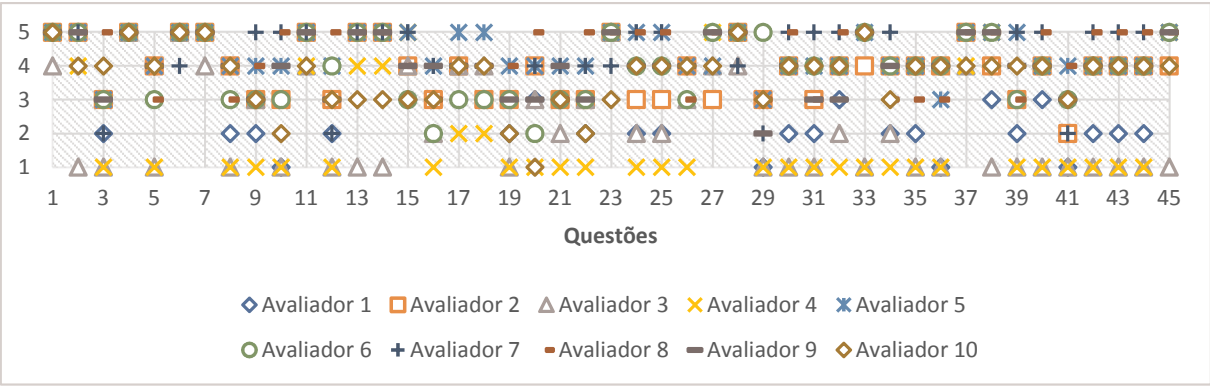


Fonte: Elaborado pelo autor.

Gráfico 27 – Avaliação da naturalidade do conjunto de Sentenças SIP.

Fonte: Elaborado pelo autor.

Gráfico 28 – Avaliação da naturalidade do conjunto de Sentenças SIL.



Fonte: Elaborado pelo autor.

Tabela 3 – Valores das métricas para do conjunto de avaliações de todos os participantes.

Dimensão de avaliação	Conjunto de Sentenças	Métricas estatísticas	Valores encontrados
Gramática	SAP	Média das notas	4,635
		Mediana	4,9
		Desvio-padrão	0,470
		Concordância	0,246
	SAL	Média das notas	4,064
		Mediana	4,1
		Desvio-padrão	0,606
		Concordância	0,163
	SIP	Média das notas	4,584
		Mediana	4,8
		Desvio-padrão	0,510
		Concordância	0,124
	SIL	Média das notas	4,033
		Mediana	3,9
		Desvio-padrão	0,690
		Concordância	0,278
Naturalidade	SAP	Média das notas	3,919
		Mediana	4
		Desvio-padrão	0,586
		Concordância	0,058
	SAL	Média das notas	3,447
		Mediana	3,5
		Desvio-padrão	0,675
		Concordância	0,073
	SIP	Média das notas	3,616
		Mediana	3,8
		Desvio-padrão	0,654
		Concordância	0,035
	SIL	Média das notas	3,616
		Mediana	3,4
		Desvio-padrão	0,755
		Concordância	0,133

Fonte: Elaborado pelo autor.

6.3.5 Análise dos resultados

Nesta seção serão analisados e discutidos os resultados das avaliações. Antes de iniciar as discussões é importante considerar que muitos avaliadores ficaram tendenciosos em suas avaliações gramaticais. Muitos responderam à pesquisa

salientando que não puderam separar a avaliação gramatical sem considerar a semântica da frase. Contudo, algumas sentenças que estavam corretas gramaticalmente tiveram notas baixas, pois não tinham sentido semântico claro.

Uma das primeiras observações que podem ser feitas é que os dados da DBpedia estão desatualizados. Logo, as sentenças podem conter informações imprecisas. No entanto, esta dimensão não está sendo avaliada, pois entende-se que ela transcende a responsabilidade do algoritmo na geração das sentenças. Essa responsabilidade é dos mantenedores da base de dados aberta e conectada que é fonte do algoritmo para geração de sentenças.

No entanto, o algoritmo teve resultado promissor na geração de linguagem natural utilizando os recursos da BDAC. A média das notas, conforme Tabela 3, foi elevada para a dimensão gramatical, acima da nota 4, que pode ser considerada como sentenças de boa construção gramatical. Já a dimensão naturalidade teve médias acima de 3,4, nota que pode ser considerada como construções aceitáveis, mas que devem ser melhoradas para alcançar melhores sentenças. Para análise da naturalidade, é importante frisar que os avaliadores nativos e não-nativos são profissionais da área da linguística. Desta forma, há um ponto que deve ser levado em consideração na interpretação deste resultado. Os avaliadores, além de ser profissionais da área, também têm diversidade de opinião e podem ter um grau de exigência maior no que se considera natural. Principalmente os avaliadores nativos.

Neste contexto, nas próximas subseções serão apresentadas análises detalhadas de cada um dos conjuntos. Optou-se por destacar e discutir os principais pontos de falha para construção das sentenças com notas mais baixas. Neste sentido, para esta análise, optou-se por apresentar as três sentenças com piores notas e explorar os motivos pelos quais elas perderam pontos junto aos avaliadores. Na maioria dos casos a escolha das três piores sentenças sumarizam e/ou evidenciam os principais problemas encontrados nos conjuntos de sentenças. Ou seja, o problema de uma sentença tende a se repetir em outras sentenças entre os conjuntos.

6.3.5.1 Gramática do conjunto de sentenças afirmativas para entidades da classe pessoa (SAP)

Avaliando com mais detalhes cada um dos conjuntos de respostas, é possível encontrar alguns motivos pelos quais as sentenças perderam pontos durante sua

avaliação. Iniciando pela avaliação gramatical do conjunto SAP, em todos os conjuntos de respostas (CR1, CR2 e CR3), é possível observar que as sentenças S3, S10, S25 tiveram notas mais baixas e essas notas foram semelhantes entre os falantes nativos e não-nativos.

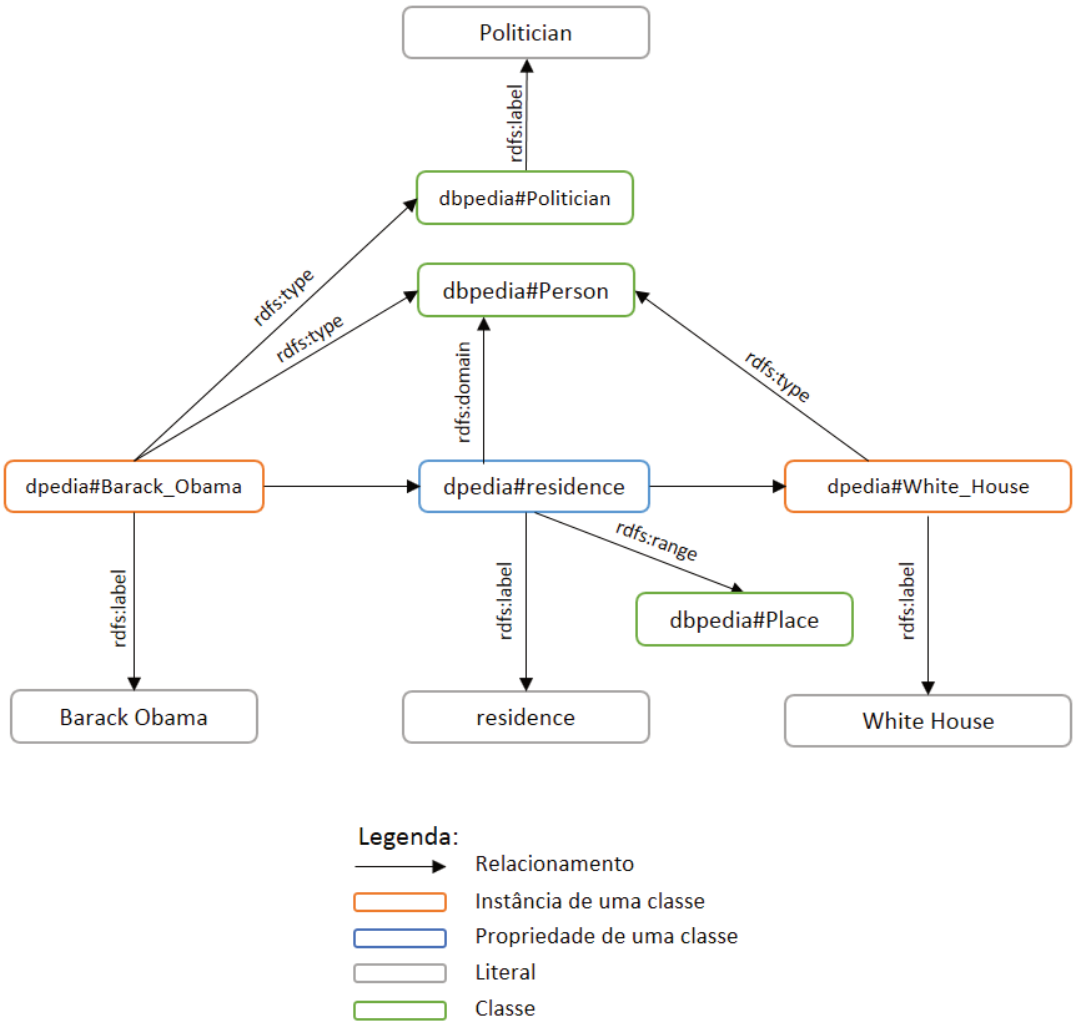
Quadro 13 – As três sentenças com médias mais baixas para a gramática (SAP).

ID	Sentença	Média
S3	Barack Obama's residence is White House.	3,6
S10	Republican Party (United States) has Hillary Clinton as a member of their other party.	3,6
S25	Ben Affleck's birth places are Berkeley, California.	3,2

Fonte: Elaborado pelo autor.

As sentenças listadas no Quadro 13 foram as sentenças com as médias mais baixas dentre os avaliadores e elas estão realmente mal construídas do ponto de vista gramatical. S3 precisa do artigo definido *The* na frente do objeto para estar correta do ponto de vista gramatical. O esquema apresentado na Figura 7 ilustra o subgrafo contendo os dados utilizados pelo algoritmo para geração desta sentença. É possível observar que, com os valores retornados da DBpedia, o algoritmo compôs a sentença sem a presença do artigo. Para que isso ocorresse, ou o template escolhido pelo algoritmo em tempo de execução deveria prever um artigo antes do objeto ou a instância que representa a Casa Branca na DBpedia deveria conter o artigo *The* em seu *label*. O mais adequado neste cenário seria a instância conter o nome correto da entidade representada.

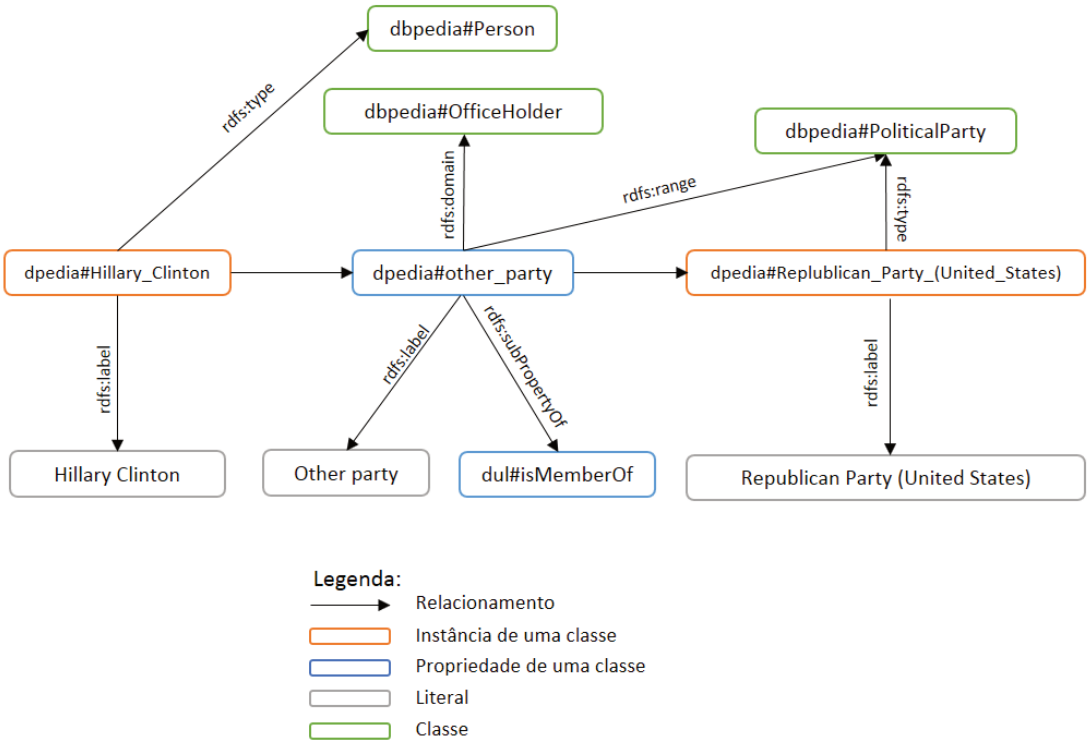
Figura 7 – Subgrafo contendo a tripla que gerou a sentença S3.



Fonte: Elaborado pelo autor.

A sentença S10 teve uma nota baixa, pois o *label* do predicado da relação entre a instância do partido político de oposição da pessoa não tem uma descrição indicativa do significado que a relação representa. Se o *label* estivesse melhor descrito, acredita-se que a construção seria mais adequada do ponto de vista gramatical (Figura 8).

Figura 8 – Subgrafo contendo a tripla que gerou a sentença S10.



Fonte: Elaborado pelo autor.

A sentença S25 também está incorreta gramaticalmente, mas por um motivo diferente. Nessa sentença o algoritmo utilizou uma tripla que continha um objeto multivalorado. Neste caso, ele utilizou o template para objetos multivalorados. O problema, no entanto, foi que o segundo objeto não passou nas validações do algoritmo, pois ele continha uma palavra já presente no primeiro objeto.

Nestes três casos, que tiveram uma nota inferior por parte dos avaliadores, há algumas soluções que podem ser comentadas. Para o primeiro cenário, há duas abordagens que podem ser levadas em consideração a fim de evitar essa situação em uma nova versão do algoritmo. A primeira seria garantir que a base de dados aberta e conectada, que é fonte do algoritmo, contenha dados corretos.

Já para o segundo cenário uma solução seria descrever melhor o recurso (predicado da tripla) que relaciona as duas entidades na base de dados para que a tripla possa ser melhor lexicalizada.

O terceiro cenário também pode ser corrigido através da manutenção dos dados presentes na BDAC. No entanto, o algoritmo de validação do algoritmo pode ser revisto para que este cenário não ocorra.

No entanto, mesmo algumas sentenças estando um pouco longe do ideal, as métricas estatísticas das notas das sentenças do conjunto SAP foram as melhores dentre os conjuntos avaliados, como pode ser observado no Tabela 3. A média de notas foi 4,6, com uma mediana de 4,9 e desvio-padrão de 0,47. Isto indica uma variância baixa das notas das sentenças, reforçando a qualidade do texto produzido para este conjunto de sentenças. Consequentemente, a concordância entre os avaliadores obteve o coeficiente de Kappa de 0,25 que pode ser interpretado como uma concordância justa entre os avaliadores (Tabela 3).

6.3.5.2 Gramática do conjunto de sentenças afirmativas para entidades da classe lugar (SAL)

No conjunto SAL as sentenças já destoam dos valores encontrados no conjunto SAP na dimensão gramatical de avaliação. Começando pela média das notas que foi de 4,06. Diferentemente do primeiro conjunto, observa-se que a mediana é próxima da média das notas e o desvio-padrão é de 0,6. Esses valores indicam uma leve variação das notas em torno do conceito bom (nota 4). Isso também é facilmente visível no Gráfico 22, onde os pontos se concentram entre os eixos de 3 a 5 na avaliação. Essa mesma concentração de notas entre 3 e 5 pode ser observada quando comparada os falantes nativos e não-nativos. Essa mesma figura mostra também certa discordância entre os avaliadores. Isso também é confirmado pelo coeficiente de Kappa para este conjunto que ficou em 0,16, que pode ser interpretado como uma concordância leve entre os avaliadores.

O Quadro 14 contempla as três sentenças com os piores desempenhos do conjunto. Em uma primeira análise é possível observar que duas das sentenças envolvem relações entre instâncias e tipos numéricos de dados. Isso já fornece alguns indicativos como o fato de que predicados deste tipo podem ser mais complicados de gerar sentenças legíveis dado a necessidade conversão e/ou formatação dos valores-objeto da sentença.

Quadro 14 – As três sentenças com médias mais baixas para a gramática (SAL).

ID	Sentença	Média
S2	Russia has a area total (m2)S of 1.70752e+13	3,1
S19	United States, whose official language is Federal government of the United States, is a country.	3,3
S30	Chicago is a place whose are code is 312/872 and 773/872.	3,2

Fonte: Elaborado pelo autor.

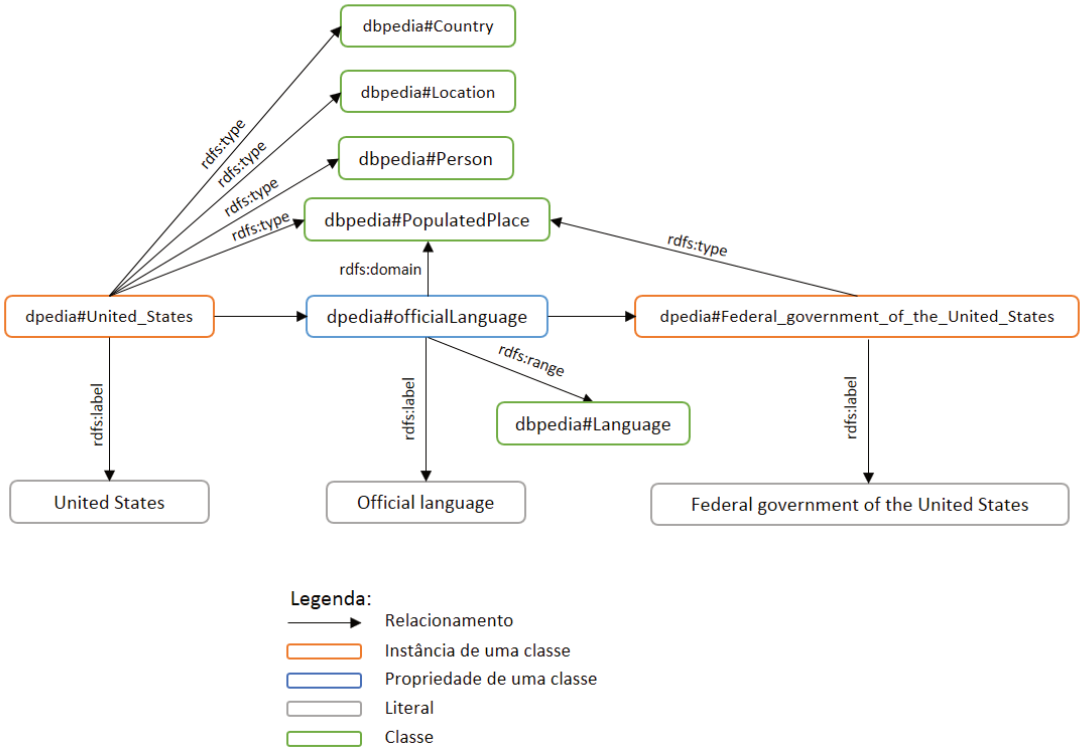
Quadro 15 – As três sentenças com médias mais altas para a gramática (SAL).

ID	Sentença	Média
S3	Russia's capital is Moscow.	5,0
S9	Canada's largest city is Toronto.	5,0
S40	Porto Alegre's area code is +55 51.	5,0

Fonte: Elaborado pelo autor.

As sentenças S2 e S30 são claros exemplos de valores numéricos nos quais o algoritmo tentou gerar uma sentença e ela não ficou perto do ideal. Já na sentença S19 é visível que houve uma captura incorreta do predicado referente a língua oficial dos EUA. A Figura 9 representa um esquema da construção da sentença 19.

Figura 9 – Subgrafo contendo a tripla que gerou a sentença S19.



Fonte: Elaborado pelo autor.

Para estas sentenças, além da formatação do numeral para um formato mais amigável, também deveria haver uma forma de identificar na ontologia a unidade de medida para que fosse possível apresentá-lo de forma mais legível. Como o algoritmo não previa tais comportamentos, numerais extensos e predicados que contém unidade de medida em seus rótulos, ele acabou gerando, neste e nos demais conjuntos, sentenças que obtiveram notas baixas.

Para a sentença S19, observa-se que o resultado da tripla não é um idioma, como seria o esperado para construção da sentença fazer sentido. O objeto da tripla é a instância representativa do governo federal dos EUA. Neste caso, o problema é claramente os dados presentes nesta tripla que resultaram em uma sentença incoerente.

Com essas avaliações é possível identificar um pouco melhor o comportamento deste conjunto. Conforme já mencionado, o sujeito de uma tripla tem uma classe que tem diversas propriedades (relacionamentos). Esses relacionamentos se repetiram pelas instâncias escolhidas por pertencerem às mesmas classes. Logo, os mesmos

problemas puderam ser identificados em outras sentenças geradas para este conjunto.

No entanto, às vezes, esse mesmo predicado tem um desempenho elevado para outras instâncias. Esse fato pode ser observado claramente entre as sentenças S30 e S40 (Quadro 15). O mesmo predicado com templates distintas obtiveram um resultado baixo e elevado para diferentes instâncias. Isso se deve, principalmente, ao fato de como o dado está apresentado na DBpedia.

6.3.5.3 Gramática do conjunto de sentenças interrogativas para entidades da classe pessoa (SIP)

A avaliação gramatical do conjunto SIP obteve o segundo lugar na avaliação geral dos indicadores estatísticos. A média das notas foi de 4,58. Semelhante ao conjunto SAP, a mediana foi bem próxima da nota máxima e o desvio-padrão foi de 0,51. No entanto, diferentemente do primeiro conjunto, o coeficiente de concordância indiciou uma concordância leve entre os avaliadores. Todos esses indicadores podem ser visivelmente identificados no Gráfico 23. Na figura há algumas sentenças onde houve discordância entre os avaliadores em relação a nota, mas outros pontos claros de concordância. Nesse conjunto de sentenças, outra observação clara, não observada nos conjuntos de sentenças discutidos anteriormente, é a discordância entre os avaliadores nativos e não-nativos na avaliação das sentenças com notas mais baixas.

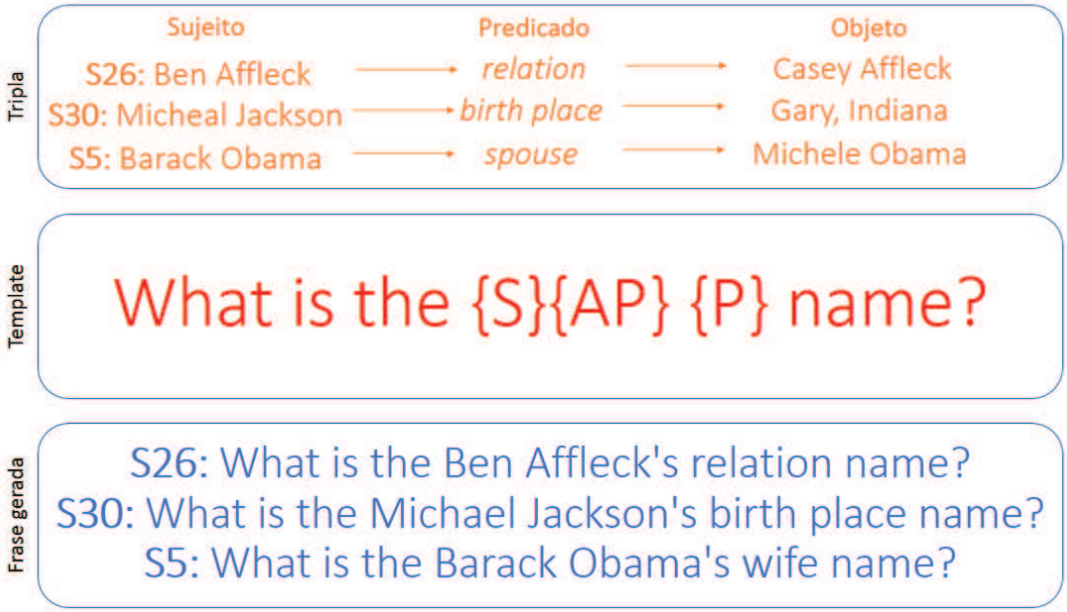
Seguindo a lógica das demais avaliações apresentadas até aqui, o Quadro 16 contém as três sentenças com as avaliações mais baixas.

Quadro 16 – As três sentenças com médias mais baixas para a gramática (SIP).

ID	Sentença	Média
S26	What is the Ben Affleck's relation name?	3,3
S30	What is the Michael Jackson's birth place name?	3,3
S5	What is the Barack Obama's wife name?	3,5

Fonte: Elaborado pelo autor.

Figura 10 – Esquema representativo do template utilizado para as sentenças.



Fonte: Elaborado pelo autor.

Diferentemente das sentenças com notas mais baixas dos conjuntos anteriores, observa-se que as sentenças com as notas mais baixas foram geradas a partir de um único *template*. Ou seja, o ponto de falha comum desse conjunto de sentenças foi um único *template* que não ficou bem construído e ocasionou esse problema (Figura 10). A situação que resultou nas sentenças do Quadro 16 também ocorreu em outras sentenças. Essa ocorrência fez com que a nota geral do conjunto SIP tivesse um desempenho mais baixo.

Além desse ponto de falha, também é notável outro problema no uso de recursos por parte do algoritmo. Na sentença S26, a tripla da relação de parentesco entre duas instâncias do tipo pessoa não pôde ser convertida com sucesso para uma sentença adequada em linguagem natural. O *label* do predicado não tem uma palavra clara indicativa deste sentido.

A partir dessas análises, possíveis resoluções seriam, primeiramente, ajustar o *template* para não incluir o artigo entre o verbo e o objeto de questionamento. Outro ajuste seria garantir que os *labels* dos predicados contenham palavras que representem claramente a relação expressada por ele. O primeiro ajuste é algo que pode ser modificado nas configurações do algoritmo. Basta ajustá-lo ou removê-lo do conjunto de *templates* para este tipo de sentença. O segundo não depende do

algoritmo. O ajuste deve ser feito diretamente na ontologia da base de dados, no caso, a DBpedia.

6.3.5.4 Gramática do conjunto de sentenças interrogativas para entidades da classe lugar (SIL)

A gramática para as sentenças do conjunto SIL teve resultados promissores. Sua média de notas foi de 4 (bom), com uma mediana muita próxima desse valor. O coeficiente de Kappa encontrado indica uma concordância justa entre os avaliadores.

Quadro 17 – As três sentenças com médias mais baixas para a gramática (SIL).

ID	Sentença	Média
S41	What is the leader titles of Chicago?	2,6
S19	What is the leader titles of United States?	2,9
S22	What is the Virginia's capital name?	3,0

Fonte: Elaborado pelo autor.

Nas sentenças S41 e S19 do conjunto SIL é possível observar um problema no *template* para sentenças interrogativas que têm mais de um objeto (Quadro 17). O *template* não previu a flexão do verbo para objetos multivalorados. Esse fator fez com que o conjunto de sentenças tivesse um desempenho mais baixo. Fica claro, mais uma vez, que o *template* para sentenças interrogativas precisa ser ajustado. Esse mesmo *template* deve ser flexível o suficiente para prever triplas com objetos multivalorados e também ter alguma inteligência para incluir ou não o artigo definido '*The'* no início de suas construções. Esse último seria suficiente para prevenir que o problema na sentença S22 ocorresse. Além disso, essa inclusão incorreta do artigo definido também foi notada no conjunto SIP.

6.3.5.5 Naturalidade do conjunto de sentenças afirmativas para entidades da classe pessoa (SAP)

O conjunto SAP novamente obteve a melhor avaliação só que agora na dimensão naturalidade. A média, no entanto, foi um pouco mais baixa da média da avaliação gramatical. Obteve-se 3,92 de média para esse conjunto com uma mediana de 4. O desvio-padrão de 0,58 também foi considerado de leve para moderado. Já a

concordância entre os avaliadores foi consideravelmente mais baixa da encontrada na avaliação da dimensão gramatical. O coeficiente ficou em 0,06 indicando uma concordância leve entre os avaliadores (Tabela 3). Isso também ocorreu entre os avaliadores nativos e não-nativos. As notas das sentenças para os dois grupos foram diversificadas, o que resultou que o coeficiente de Kappa para este conjunto fosse o mais baixo dentre os estudados.

Quadro 18 – As três sentenças com médias mais baixas para a naturalidade (SAP).

ID	Sentença	Média
S10	Republican Party (United States) has Hillary Clinton as a member of their other party.	2,5
S12	Hillary Clinton's presidents are Bill Clinton and Barack Obama.	2,7
S30	Ben Affleck's relation is Casey Affleck.	2,9

Fonte: Elaborado pelo autor.

A sentença S10, novamente para o conjunto SAP, teve uma nota baixa. Desta vez, mesmo com outro *template*, a sentença não foi bem construída em função do *label* do predicado da relação entre a instância da pessoa política e seu partido. O *label* do partido não tem uma descrição indicativa do significado que a relação representa. Nesta sentença específica, o template é até um pouco mais complexo, indicando o pertencimento da instância ao objeto da relação. Se o *label* estivesse mais bem descrito, talvez o resultado desta construção fosse uma sentença mais legível e, por consequência, mais natural aos olhos dos avaliadores.

Na sentença S12 a palavra presidente está sendo utilizada em outro sentido. Um sentido não natural para os avaliadores. Ela significa que a entidade representada na instância participou como servidora pública de dois mandatos de presidente. A palavra presidente, neste caso, tem sentido de chefe e não de presidente. No entanto, a sentença falha em expressar essa informação e foi motivo de notas mais baixas por parte dos avaliadores.

Já a sentença S30 também apareceu na avaliação gramatical. Novamente em função da palavra que descreve o predicado da tripla escolhida não ser a ideal para representação do significado da relação.

Para todos os casos acima que tiveram as notas mais baixas na avaliação da naturalidade, todas as construções falharam na expressão de seus significados em

função da descrição das relações fornecidas pela DBpedia. Novamente isso reforça a hipótese de que se a base estivesse melhor descrita, esse tipo de sentença não existiria e a avaliação, tanto das sentenças do Quadro 18, quanto as demais, tivessem notas mais elevadas.

6.3.5.6 Naturalidade do conjunto de sentenças afirmativas para entidades da classe lugar (SAL)

A avaliação da naturalidade do conjunto SAL obteve a nota mais baixa nessa dimensão dentro dos conjuntos de sentenças avaliadas. Sua média foi de 3,45 com uma mediana não muito longe desta média, 3,5. O desvio padrão foi considerado moderado com um valor de 0,68. Além do mais, houve concordância entre os avaliadores, mas ela foi próxima de 0. Isso é claramente notado no Gráfico 26. Também é possível observar que entre nativos e não-nativos a baixa concordância na avaliação das sentenças também existiu (Gráficos 11 e 19).

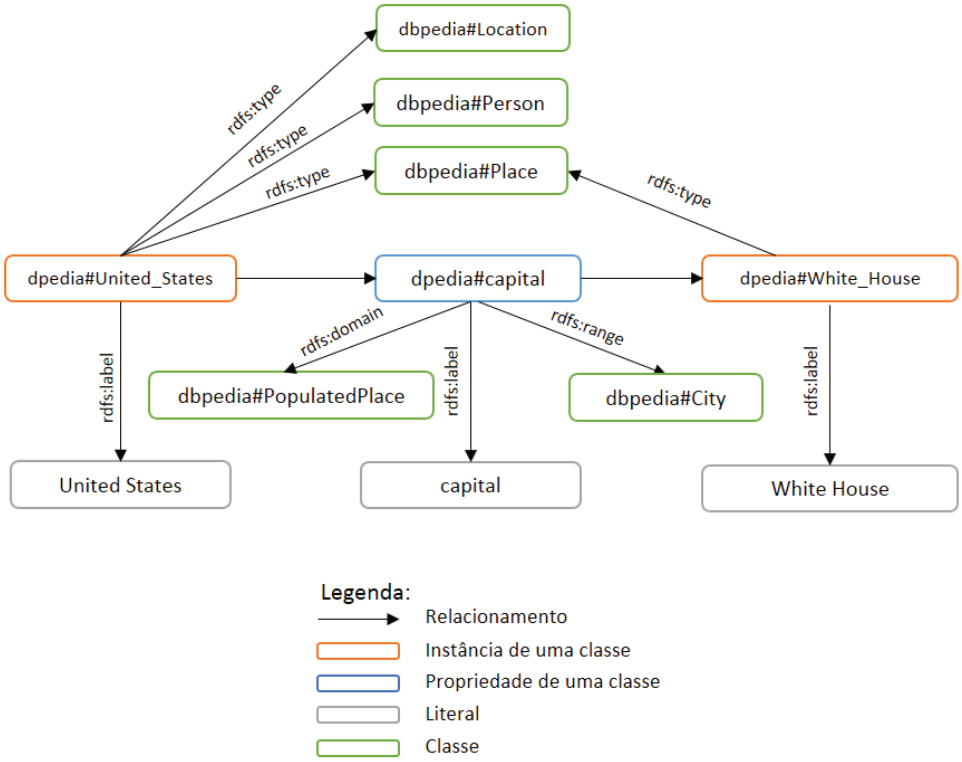
De fato, analisando a Tabela 19 contendo as 3 sentenças com as notas mais baixas dentre as sentenças avaliadas também é possível perceber que essas três sentenças estão entre as sentenças com notas mais baixas em relação ao conjunto total de sentenças.

Quadro 19 – As três sentenças com médias mais baixas para a naturalidade (SAL).

ID	Sentença	Média
S20	United States, whose capital is Washington - D.C., is a person.	2,1
S19	United States, whose official language is Federal government of the United States, is a country.	2,2
S14	Canada's section is Ottawa.	2,4

Fonte: Elaborado pelo autor.

Figura 11 – Subgrafo contendo a tripla que gerou a sentença S20.



Fonte: Elaborado pelo autor.

Iniciando pela sentença S20, podemos perceber que houve um erro considerável na semântica da sentença. A Figura 11 mostra quais foram os elementos que o algoritmo utilizou para gerar a sentença. O *template* escolhido pelo algoritmo em tempo de execução previa a seleção de uma das classes do sujeito. Neste caso, a instância *United States* foi consultada para identificar suas classes (*rdf:type*). Havia um conjunto de classes e o algoritmo escolheu uma a partir da aleatoriedade. No entanto, ele teve o infortúnio de escolher a classe pessoa para esta instância que representa um local. Nota-se que o problema, neste caso, não é originado pelo algoritmo, já que ele executou o correto: escolheu uma classe da instância. O problema é uma instância que representa um local ter associado a ela uma classe que representa uma pessoa. Esse erro de relação presente na DBpedia levou o algoritmo a criar uma sentença incorreta do ponto de vista semântico e, consequentemente, não-natural para os avaliadores.

A sentença S19 já apareceu na avaliação gramatical do conjunto SAL e ela tem um problema parecido com a sentença S20 descrita acima. A relação que representa

o idioma oficial dos EUA contém um dado incorreto e que não tem o sentido que a relação representa. Desta forma, as notas dos avaliadores foram baixas.

Já a sentença S14 tem um problema diferente. O algoritmo não teve sucesso ao substituir a palavra '*capital*' por um sinônimo mais frequente e natural. Ele escolheu uma palavra pouco usual para representar essa relação. Analisando um pouco mais o motivo pelo qual essa escolha ocorreu, identificou-se que ao consultar o WordNet, recebeu uma palavra com o significado diferente do original. Por essa palavra, ele consultou um dicionário de sinônimos que retornou uma palavra totalmente diferente da esperada.

A resolução dos problemas evidenciados nas sentenças do Quadro 19 e presentes em outras sentenças que também tiveram notas baixas novamente recaem na gestão das informações presentes na DBpedia. Novamente o algoritmo foi vítima de descrições incoerentes para o real significado da relação. Também se evidenciou, principalmente pela sentença S14, que somente um mapeamento de agente semântico entre as classes da DBpedia e WordNet não foram suficientes para garantir a escolha de sinônimos que, pelo menos, pertencessem ao mesmo grupo semântico. Esse ponto pode ser explorado com mais ênfase em pesquisas futuras como será descrito na seção final deste trabalho.

6.3.5.7 Naturalidade do conjunto de sentenças interrogativas para entidades da classe pessoa (SIP)

Esse conjunto conta com a sentença com a média mais baixa na avaliação geral de naturalidade de todas as sentenças (Sentença S21). A média geral deste conjunto foi de 3,61, com uma mediana de 3,8, próxima da média, e o desvio-padrão foi de 0,65. A concordância também foi baixa entre os avaliadores. Isso pode ser visto pelo Gráfico 27 e também pelo valor do coeficiente de Kappa que foi de 0,035. A concordância entre os avaliadores nativos e não-nativos também foi leve, como pode ser observado nos Gráficos 10 e 18.

Quadro 20 – As três sentenças com médias mais baixas para a naturalidade (SIP).

ID	Sentença	Média
S21	Cite at least one of Ben Affleck's birth places.	2,0
S26	What is the Ben Affleck's relation name?	2,4
S19	Cite at least one of Hillary Clinton's president.	2,6

Fonte: Elaborado pelo autor.

No entanto, no Quadro 20 encontram-se as sentenças com as avaliações mais baixas. Observa-se a concordância entre os avaliadores ao avaliar com notas mais baixas estas sentenças, tanto entre os nativos, quanto dos não-nativos (Gráficos 10 e 18). Destaca-se a sentença interrogativa S21 para relação da instância que representa o ator Ben Affleck e seu local de nascimento. Novamente, em virtude desta tripla ser multivalorada, o algoritmo escolheu um template para este fim. No entanto, uma pessoa não tem dois locais de nascimento. Logo, a sentença perdeu pontos consideráveis em virtude dessa incoerência promovida pelos dados disponíveis na DBpedia para essa relação.

A sentença S26 também teve um problema com a instância do ator Ben Affleck. Também é uma tripla que já apresentou problemas na avaliação gramatical deste mesmo conjunto. A palavra *relation* não indica o real significado da relação expressa pela tripla. Deste modo a sentença destoa muito daquilo que é entendido como natural.

Com o mesmo problema das demais, a sentença S19, também já destacada na avaliação da dimensão gramatical de outros conjuntos de sentenças, não representa o real significado da relação expressada pela tripla. Conforme mencionado anteriormente, a palavra presidente neste caso tem o sentido de chefe e não de presidente, uma vez que o sujeito da tripla foi servidora na gestão dos dois presidentes. Uma sentença mais elaborada poderia expressar esse significado, mas com os recursos disponíveis ao algoritmo em tempo de execução, ele não conseguiu realizar essa tarefa.

6.3.5.8 Naturalidade do conjunto de sentenças interrogativas para entidades da classe pessoa (SIL)

Por fim, a avaliação da naturalidade do conjunto SIL teve uma média muito próxima das demais avaliações de naturalidade. Sua média foi 3,5 com mediana de 3,3. No entanto, esse conjunto teve o maior desvio-padrão dos conjuntos avaliados. O desvio-padrão foi de 0,78 pontos. A concordância entre os avaliadores também foi a mais baixa dentre todos os conjuntos de sentenças analisadas. O coeficiente foi próximo de zero, mas, ainda assim, indica certa concordância entre os avaliadores. Isso pode ser notado quando observado todos os participantes e entre os avaliadores nativos e não-nativos (Gráficos 12, 20 e 28).

Quadro 21 – As três sentenças com médias mais baixas para a naturalidade (SIL).

ID	**Sentença**	**Média**
S29	What is Porto Alegre's cartesian product?	2,4
S41	What is the leader titles of Chicago?	2,4
S19	What is the leader titles of United States?	2,5

Fonte: Elaborado pelo autor.

O Quadro 21 contempla as três sentenças com as avaliações mais baixas. É possível perceber que duas delas (S41 e S19) tiveram a raiz do problema em uma situação já comentada anteriormente (Figura 38). A inflexão do verbo em uma sentença interrogativa para uma tripla com objeto multivalorado. Novamente o problema recai no template escolhido que apresenta inconformidade na construção de sentenças gramaticalmente corretas. E estas, por sua vez, tendem a ter uma avaliação de naturalidade mais baixa.

Além disso, outro problema também já comentado é a escolha do sinônimo. Para a sentença S29 o sinônimo encontrado para a descrição do predicado '*area code*' que representa o DDI e DDD da localidade teve como resultado '*cartesian product*'. Essa escolha incorreta teve como origem o atributo de frequência dos sinônimos de '*area code*' no léxico utilizado pelo algoritmo. Infelizmente, a palavra retornada não compartilha do mesmo significado que a palavra pesquisada. A resolução deste problema segue na mesma linha que já foi comentado anteriormente, uma

configuração mais detalhada relacionamentos entre as classes da DBpedia e as informações presentes no WordNet.

6.3.6 Discussão dos resultados

De forma geral, é possível observar uma indicação positiva de geração de linguagem a partir dos métodos propostos neste trabalho. As sentenças geradas tiveram mais avaliações positivas que negativas. Houve concordância entre os avaliadores a partir da interpretação do coeficiente de Kappa e dos valores de mediana e desvio padrão dos conjuntos estudados.

No entanto, o algoritmo teve insucesso em algumas situações. São essas situações que precisaram ser exploradas e analisadas com maior detalhe para corrigir e reexecutar as avaliações. Logo, nos próximos parágrafos, serão discutidos os principais pontos de falha encontrados durante a análise dos resultados.

Um dos principais fatos observados é que o algoritmo teve um maior êxito na dimensão gramatical do que na dimensão natural de todos os conjuntos de sentenças.

A partir da discussão da seção anterior, é possível observar que há três itens que foram a causa raiz dos problemas apresentados na maioria das sentenças com notas mais baixas. O primeiro foi a qualidade da informação dos itens que compõe uma tripla dentro da DBpedia. A maioria está descrita de uma forma não legível para o modo de operação do algoritmo. Elas necessitam de um tratamento adicional por parte do algoritmo para se tornarem mais legíveis. Isso foi algo não identificado claramente nas primeiras avaliações e só notado nesta última avaliação.

Ainda neste item, há um exemplo claro deste problema. As instâncias da classe Pessoa tiveram menos relações selecionadas para lexicalização do que as instâncias da classe Lugar. No entanto, os objetos das relações das instâncias da classe Lugar eram os que tinham dados com a menor qualidade. Observa-se na avaliação dos conjuntos de sentenças que as sentenças afirmativas para as instâncias da classe Lugar tiveram notas mais baixas que as sentenças interrogativas dessas mesmas instâncias, pois o objeto, em sentenças interrogativas, é oculto.

O segundo item que pode ser identificado nas sentenças com notas mais baixas foi o uso de alguns *templates* que não estavam completamente refinados para alguns contextos. Logo, uma tripla que atendia os requisitos mínimos do algoritmo

para geração de linguagem gerou uma sentença distante do ideal para as dimensões de avaliação gramatical e natural. No entanto, se essa mesma sentença fosse lexicalizada por outro *template*, a tendência é que ela se aproximasse de uma sentença considerada ideal pelos os avaliadores, tanto gramaticalmente, quanto naturalmente.

Por fim, outro ponto identificado é a escolha de sinônimos que não compartilhavam do mesmo significado que o rótulo de uma relação entre o sujeito e objeto de uma tripla. Em algumas situações o algoritmo escolheu, para palavras com mais um sentido semântico, um sinônimo de outro sentido semântico. Logo, para este ponto, o algoritmo necessita reconhecer o sentido semântico da tripla para buscar pelo sinônimo correto. Algo que já havia sido identificado na segunda avaliação e que a solução implementada por esta versão atual do algoritmo não conseguiu cobrir perfeitamente.

Desta forma, a partir dos resultados encontrados e análises executadas, é possível formular uma nova hipótese: o algoritmo pode obter maior êxito no seu objetivo de geração de linguagem natural se refinar os *templates* linguísticos para eliminar os que geram sentenças avaliadas com notas mais baixas e incluir uma nova etapa para tratamento dos dados obtidos da BDAC ou aplicá-lo somente em BDAC que contenham dados previamente tratados para apresentação por sistemas computacionais.

Além disso, também pode ser observado que tanto os avaliadores nativos e não-nativos tiveram concordâncias e discordâncias entre eles na avaliação das sentenças. No entanto, observou-se que os falantes nativos tendem a avaliar as sentenças com maiores problemas com notas mais baixas que os avaliadores não-nativos.

O fato é que qualquer que seja o modo que uma sentença seja gerada por computadores, ela deve considerar o público-alvo para obter um resultado mais próximo do natural para aquele público.

Contudo, o algoritmo inova pelo fato de gerar linguagem natural utilizando somente o contexto ontológico de BDAC e não consultar corpus auxiliares como alguns trabalhos relacionados.

Além disso, o algoritmo se diferencia dos demais pela aproximação com a área da linguística, uma vez que para gerar sentenças utiliza formulações linguísticas (*templates*) construídas por pesquisadores desta área de estudo.

Ademais, a construção do algoritmo da forma apresentada até aqui se pautou pela análise dos resultados de avaliações preliminares com envolvimento de falantes nativos e não-nativos do idioma inglês. A realização destas avaliações buscava a geração de sentenças naturais e gramaticalmente corretas. Algo alcançado para uma variedade de sentenças, como pode ser observado pela média geral de alguns conjuntos.

7 CONSIDERAÇÕES FINAIS

Os conhecimentos armazenados através de dados abertos e conectados (BDAC) podem ser explorados de diversas maneiras. Uma delas é o uso das informações contidas nessas bases para geração de linguagem natural. Esta pesquisa buscou gerar frases curtas utilizando as informações presentes em BDAC com apoio de *templates* linguísticos.

Diferente dos trabalhos relacionados e estudados, a partir da melhor avaliação de nosso conhecimento no tema, o algoritmo proposto é o único que explora aspectos linguísticos das principais propriedades definidas no RDF. Ao explorar esses aspectos, o algoritmo difere dos demais para geração de linguagem a partir de BDAC, pois as abordagens estudadas se limitam ao uso de poucas propriedades (*label* e *type*, por exemplo).

O algoritmo também tira proveito de uma base com mais de dez *templates* linguísticos gerados com o apoio de um grupo de pesquisadores da área da linguística, que exploraram pronomes, diferentes formas lógicas (por exemplo, as linguagens ativa e passiva) e diversos tipos de pontuação para obter *templates* que soassem naturais e corretos gramaticalmente, aspecto que não foi encontrado nos trabalhos relacionados. Também difere dos demais trabalhos atuais sobre o tema por utilizar dois recursos linguísticos adicionais. Léxicos de sinônimos e combinação de palavras foram utilizados visando aumentar a diversidade e naturalidade nas sentenças geradas

Neste contexto, o algoritmo foi modelado a partir de duas avaliações preliminares. Foi executada uma primeira avaliação para validar a possibilidade de gerar linguagem natural a partir de bases de dados abertos e conectados. A partir deste resultado, uma nova avaliação foi executada validando os ajustes no algoritmo que foram implementados para aumentar a naturalidade e diversidade nas sentenças geradas.

Nestas duas avaliações preliminares, observaram-se bons resultados. Desta forma, uma avaliação final foi elaborada após os ajustes necessários identificados nas duas avaliações preliminares. Essa avaliação final teve como avaliadores linguistas nativos e brasileiros para avaliação das frases curtas geradas pelo algoritmo. Além da média de notas nas dimensões de gramática e naturalidade, na análise dos resultados desta última avaliação, também foi avaliado a concordância dos avaliadores a partir

do estudo das métricas de desvio-padrão, mediana e coeficiente de Kappa para mais de dois avaliadores. O método de avaliação aplicado nesta pesquisa também difere dos trabalhos recentes relacionados ao tema, pois avalia a concordância entre os avaliadores.

A análise dos resultados indica que, no geral, o algoritmo teve bons resultados na geração de frases curtas. Para alguns conjuntos de sentenças ele teve média de notas que indicam sentenças perfeitas, tanto nas dimensões gramatical, quanto na natural. No entanto, para outros conjuntos de sentenças, as notas, dentro do intervalo de avaliação, foram menores.

Observou-se que o algoritmo apresentou dificuldades para geração de frases curtas para algumas entidades de determinadas classes. Ao analisar profundamente essas frases, notou-se que a maior parte dos problemas é advindo de dados mal estruturados ou imprecisos dentro da base de dados aberta e conectada estudada, no caso a DBPedia. Isso levou o algoritmo a gerar sentenças que não soassem naturais. Houve outros problemas pontuais de escolha de sinônimos, pois no léxico de sinônimos utilizado, o papel semântico de cada palavra não existia ou não estava definido.

No entanto, a média geral das notas de todas as sentenças, independentemente do tipo, foi considerada satisfatória (nota 4 em uma escala até 5 no máximo) com indicativo de leve concordância entre os avaliadores. Ou seja, mesmo com situações fora das responsabilidades do algoritmo, ele obteve sucesso na geração das sentenças. Portanto, é possível concluir que se o algoritmo for implantado conectado à dados abertos e conectados que estejam bem estruturados e precisos, as frases curtas estariam muito próximas da excelência.

7.1 CONTRIBUIÇÕES

Durante esses primeiros meses de pesquisa, observou-se, durante o levantamento bibliográfico, que o método e a geração de linguagem natural a partir das propriedades dos elementos de triplas RDF ainda não foi explorado do modo que está sendo explorado por este trabalho. Também foi possível notar que poucos trabalhos têm apoio linguístico para construção de sentenças com dados presentes em triplas. Desta forma, este trabalho fornece contribuições neste âmbito.

Esta pesquisa também inova no método de avaliação, ao utilizar métricas que avaliam a concordância dos avaliadores em torno dos objetos avaliados. Algo não encontrado nos trabalhos relacionados.

Alguns artigos sobre os resultados desta pesquisa já foram escritos e publicados e outros estão em processo de escrita e/ou publicação.

Essa pesquisa é uma expansão do trabalho desenvolvido para conclusão do curso de bacharelado do autor. Foi desenvolvido um jogo sério de perguntas e respostas que tinha como BDAC como fonte de conhecimento. Naquele momento, a DBPedia era consultada para encontrar triplas para escolher uma pergunta pré-definida para uma dada relação entre sujeito e objeto. Este trabalho inicial também foi publicado e apresentado no evento *Computer on the Beach* 2018.

Além disso, o algoritmo que utiliza as informações presentes em bases de dados abertos e conectados com apoio de recursos linguísticos para geração de frases curtas foi patenteado.

7.2 TRABALHOS FUTUROS

O algoritmo desenvolvido por essa pesquisa tem diversas aplicações e elas podem ser estudadas em trabalhos futuros. A seguir estes trabalhos futuros identificados como possibilidade são descritos.

O desenvolvimento de sistemas de perguntas e respostas para apoiar o ensino pode utilizar o algoritmo como parte do módulo de geração de linguagem. Neste caso, uma ontologia sobre um tema específico pode ser construída e populada por professores, por exemplo. Deste modo, um sistema de perguntas e respostas, apoiado pelo algoritmo para construção das frases simples, pode usar a ontologia para criar sentenças em um jogo sério interativo. Dependendo da ontologia, o jogo poderia ser aplicado para estudantes de todos os níveis de ensino.

O algoritmo ainda poderia ser utilizado por *chatbots* para geração de sentenças na consulta de ontologias específicas durante a interação com seres humanos. Por exemplo, se uma empresa tem uma ontologia sobre seu processo e produtos, um *chatbot* poderia usar o algoritmo para gerar sentenças em linguagem natural a partir das triplas desta ontologia.

Novos estudos estão sendo realizados para ampliar o conjunto de templates e também para executar a mineração de textos para encontrar *templates* para as triplas e suas propriedades de forma automática.

O algoritmo ainda pode ser expandido para gerar frases mais complexas ou ainda ampliado para agrupar as sentenças em parágrafos descritivos sobre entidades presentes em dados abertos e conectados. Isso poderia ser feito com a utilização de um conjunto de relações existentes na base de conhecimento a respeito de uma mesma entidade. Então todas as relações podem ser traduzidas para frases curtas individualmente e por fim conectadas em um parágrafo, situação na qual existe o desafio de concatenar logicamente e com os conectivos linguísticos corretos cada uma das frases com as demais.

Trabalhos futuros também podem utilizar o método de avaliação proposto neste trabalho. Desta forma, eles poderiam avaliar a concordância dos avaliadores em suas avaliações para mensurar a qualidade de texto gerado automaticamente para grupos de parágrafos, como citado.

Além disso, pesquisas futuras poderiam fornecer meios para comparar avaliações por especialistas com avaliações por métricas automatizadas.

REFERÊNCIAS

ABOUT. **DBPedia,** 2017. Disponível em: <https://wiki.dbpedia.org/about>. Acesso em: 06/06/2018.

AIRES, João Pinto Barbosa Machado. Automatic Generation of Sports News. Dissertação (Dissertação em Engenharia electrotécnica, electrônica e informática) – Faculdade de Engenharia, Universidade do Porto. Porto. 2016.

ANGELI, Gabor; PREMKUMAR, Melvin Jose Johnson; MANNING, Christopher D. Leveraging linguistic structure for open domain information extraction. In: **Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)**. 2015. p. 344-354.

BIRAN, Or; MCKEOWN, Kathleen. Domain-Adaptable Hybrid Generation of RDF Entity Descriptions. In: **Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)**. 2017. p. 306-315.

COHEN, Jacob. A coefficient of agreement for nominal scales. **Educational and psychological measurement**, v. 20, n. 1, p. 37-46, 1960.

COHEN, Jacob. Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit. **Psychological bulletin**, v. 70, n. 4, p. 213, 1968.

COJOCARU, Dragos Alexandru; TRAUSAN-MATU, Stefan. Text Generation Starting from an Ontology. In: **RoCHI**. 2015. p. 55-60.

COLE, Ronald et al. **Survey of the state of the art in human language technology**. 1997.

CRASWELL, Nick. Mean reciprocal rank. In: **Encyclopedia of Database Systems**. Springer US, 2009. p. 1703-1703.

DATAMUSE API. **DATAMUSE**, 2018. Disponível em: <http://www.datamuse.com/api/>. Acesso em: 20 de out. de 2018.

DBPEDIA Version 2014-10. **DBPedia**, 2014. Disponível em: <https://wiki.dbpedia.org/data-set-2014> Acesso em: 20 de out. de 2018.

DONEGAN, Patricia J.; STAMPE, David. The study of natural phonology**. Current approaches to phonological theory**, v. 126173, 1979.

DUMA, Daniel; KLEIN, Ewan. Generating natural language from linked data: Unsupervised template extraction. In: **Proceedings of the 10th International Conference on Computational Semantics (IWCS 2013)–Long Papers**. 2013. p. 83-94.

EXPRESS JS. **Node Foundation**, 2017. Disponível em: <https://www.expressjs.com/>. Acesso em: 20 de out. de 2018.

GATT, Albert; KRAHMER, Emiel. Survey of the State of the Art in Natural Language Generation: Core tasks, applications and evaluation. **Journal of Artificial Intelligence Research**, v. 61, p. 65-170, 2018.

GUPTA, Vishal. **A Survey of Natural Language Processing Techniques**. 2014.

HEATH, Tom; BIZER, Christian. Linked data: Evolving the web into a global data space. **Synthesis lectures on the semantic web: theory and technology**, v. 1, n. 1, p. 1-136, 2011.

HÖFFNER, Konrad et al. Survey on challenges of question answering in the semantic web. **Semantic Web**, v. 8, n. 6, p. 895-920, 2017.

FLEISS, Joseph L. Measuring nominal scale agreement among many raters. **Psychological bulletin**, v. 76, n. 5, p. 378, 1971.

INDURKHYA, Nitin; DAMERAU, Fred J. (Ed.). **Handbook of natural language processing**. CRC Press, 2010.

ISOTANI, Seiji; BITTENCOURT, Ig Ibert. **Dados Abertos Conectados: Em busca da Web do Conhecimento**. Novatec Editora, 2015.

KAFFEE, Lucie-Aimée et al. **Mind the (Language) Gap**: Generation of Multilingual Wikipedia Summaries from Wikidata for ArticlePlaceholders, 2018.

KITCHENHAM, Barbara; CHARTERS, Stuart. Guidelines for performing systematic literature reviews in software engineering. **EBSE Technical Report,** p. 1-57, 2007.

LANDIS, J. Richard; KOCH, Gary G. The measurement of observer agreement for categorical data. **biometrics**, p. 159-174, 1977.

LEHMANN, Jens et al. DBpedia–a large-scale, multilingual knowledge base extracted from Wikipedia. **Semantic Web**, v. 6, n. 2, p. 167-195, 2015.

LIANG, Shao Fen et al. OntoVerbal: a Protégé plugin for verbalising ontology classes. In: **ICBO**. 2012.

LIN, Chin-Yew. Rouge: A package for automatic evaluation of summaries. **Text Summarization Branches Out**, 2004.

LINKED Data Design Issues. **W3C**: World Wide Web Consortium, 2006. Disponível em: <https://www.w3.org/DesignIssues/LinkedData.html>. Acesso em: 20 de out. de 2018.

LINKED Data. **W3C**: World Wide Web Consortium, 2015. Disponível em: <https://www.w3.org/standards/semanticweb/data/>. Acesso em: 20 de out. de 2018.

LIU, Chia-Wei et al. How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. **arXiv preprint arXiv:1603.08023**, 2016.

MANN, William C.; THOMPSON, Sandra A. Rhetorical structure theory: Description and construction of text structures. In: **Natural language generation**. Springer, Dordrecht, 1987. p. 85-95.

MCCRAE, John P. **The Linked Open Data Cloud,** 2018. Disponível em: <https://lod-cloud.net/>. Acesso em: 06/06/2018.

MCKEOWN, Kathleen R.; RADEV, Dragomir R. Collocations. **Handbook of Natural Language Processing. Marcel Dekker**, 2000.

MILLER, George A. WordNet: a lexical database for English. **Communications of the ACM**, v. 38, n. 11, p. 39-41, 1995.

MOUSSALLEM, Diego et al. RDF2PT: Generating Brazilian Portuguese Texts from RDF Data. **arXiv preprint arXiv:1802.08150**, 2018.

NOVIKOVA, Jekaterina et al. Why we need new evaluation metrics for nlg. **arXiv preprint arXiv:1707.06875**, 2017.

ONTOLOGIES. **W3C**: World Wide Web Consortium, 2015. Disponível em: <https://www.w3.org/standards/semanticweb/ontology.html/>. Acesso em: 20 de out. de 2018.

OWL Web Ontology Language Guide. **W3C**: World Wide Web Consortium, 2004. Disponível em: <https://www.w3.org/TR/owl-guide/>. Acesso em: 20 de out. de 2018.

OXFORD Collocations Dictionary. **Oxford Learners Dictionary**, 2018. Disponível em: <https://www.oxfordlearnersdictionaries.com/definition/collocations/>. Acesso em: 20 de out. de 2018.

PAPINENI, Kishore et al. BLEU: a method for automatic evaluation of machine translation. **In: Proceedings of the 40th annual meeting on association for computational linguistics**. Association for Computational Linguistics, 2002. p. 311-318.

PERERA, Rivindu et al. Semantic Web Today: From Oil Rigs to Panama Papers. **arXiv preprint arXiv:1711.01518**, 2017.

PERERA, Rivindu; NAND, Parma. A multi-strategy approach for lexicalizing linked open data. In: **International Conference on Intelligent Text Processing and Computational Linguistics**. Springer, Cham, 2015. p. 348-363.

PERERA, Rivindu; NAND, Parma. Recent advances in natural language generation: A survey and classification of the empirical literature. **Computing and Informatics**, v. 36, n. 1, p. 1-32, 2017.

PERERA, Rivindu; NAND, Parma; NAEEM, Asif. Utilizing typed dependency subtree patterns for answer sentence generation in question answering systems. **Progress in Artificial Intelligence**, v. 6, n. 2, p. 105-119, 2017.

RAMOS-SOTO, Alejandro; BUGARÍN, Alberto; BARRO, Senén. On the role of linguistic descriptions of data in the building of natural language generation systems. **Fuzzy Sets and Systems**, v. 285, p. 31-51, 2016.

RDF 1.1 Concepts and Abstract Syntax. **W3C**: World Wide Web Consortium, 2014b. Disponível em: <https://www.w3.org/TR/rdf11-concepts/#section-Graph-Literal/>. Acesso em: 20 de out. de 2018.

RDF Schema 1.1. **W3C**: World Wide Web Consortium, 2014a. Disponível em: <https://www.w3.org/TR/rdf-schema/>. Acesso em: 20 de out. de 2018.

REITER, Ehud; DALE, Robert. Building applied natural language generation systems. **Natural Language Engineering**, v. 3, n. 1, p. 57-87, 1997.

RODRIGUES, Emílio Luiz Faria. Geração de perguntas em linguagem natural a partir de bases de dados abertos e conectados: um estudo exploratório. Dissertação. (Mestrado em Computação Aplicada) – Escola Politécnica, UNISINOS. São Leopoldo. 2017.

SANTOS, Irenilde Pereira dos. Lingüística. **Estudos avançados**, v. 8, n. 22, p. 481-486, 1994.

SHEKARPOUR, Saeedeh et al. Question answering on linked data: Challenges and future directions. In: **Proceedings of the 25th International Conference Companion on World Wide Web**. International World Wide Web Conferences Steering Committee, 2016. p. 693-698.

SPARQL Query Language for RDF. **W3C**: World Wide Web Consortium, 2008. Disponível em: < https://www.w3.org/TR/rdf-sparql-query/>. Acesso em: 20 de out. de 2018.

STAMPE, David. **A dissertation on natural phonology**. 1979. Tese de Doutorado. Indiana University Linguistics Club.

TECHNICAL Architecture Modeling. **SYBASE**, 2013. Disponível em: <http://infocenter.sybase.com/help/index.jsp?topic=/com.sybase.infocenter.dc38086.1652/doc/html/rad1366715674186.html> Acesso em: 15 de nov. de 2018.

UNGER, Christina; FREITAS, André; CIMIANO, Philipp. An introduction to question answering over linked data. In: **Reasoning Web International Summer School**. Springer, Cham, 2014. p. 100-140.

VICENTE, Marta et al. La generacion de lenguaje natural: análisis del estado actual. **Computación y Sistemas**, v. 19, n. 4, p. 721-756, 2015.

VOUGIOUKLIS, Pavlos et al. Neural Wikipedian: Generating Textual Summaries from Knowledge Base Triples. **arXiv preprint arXiv:1711.00155**, 2017.

WAZLAWICK, Raul. **Metodologia de pesquisa para ciência da computação**. Elsevier Brasil, 2017.

WEB Semântica. **W3C Brasil**: World Wide Web Consortium Escritório Brasil, 2011. Disponível em: <http://www.w3c.br/Padroes/WebSemantica/>. Acesso em: 20 de out. de 2018.

WEISSER, Martin. Naturalness and Spoken Data. In Kyratzis, S. & Banerjee, J. (Eds.). (2001). **Data in Applied Linguistics**. Lancaster: Department of Linguistics and Modern English Language, Lancaster University, 1997.

XAVIER, M. F.; MATEUS, M. H. Dicionário de Termos Linguísticos, II. **Associação Portuguesa de Linguística/Instituto de Linguística Teórica e Computacional (Lisboa: Cosmos)**, 1992.

ZAVAGLIA, Claudia. Base de Conhecimento Léxico-Ontológico para o Português do Brasil: uma proposta de modelo. 2003.

APÊNDICE A – TEMPLATES UTILIZADAS PELO ALGORITMO

Tipo de dado do objeto	Subtipo de dado	Sentenças afirmativas	Sentenas interrogativas
Numérico	Genérico	{S}{AP} {P} has {O}.	{PRON} is {S}{AP} {P}?
		{S} has {A} {P} of {O}	
	Ano	{S}{AP} {P} was {O}.	Which year was {S}{AP} {P}?
			Which year did {S} {PV}?
Textual	Genérico	{S}{AP} {P} {VTB} {O}.	What is the {P} of {S}?
			{PRON} is {S}{AP} {P}?
			{PRON} did {S} {PV} {OT}
	URI	{S} is {A} {ST} whose {P} {VTB} {O}.	What is the name of {S}{AP} {P}?
		{S}, whose {P} {VTB} {O}, is {A} {ST}	What is the {S}{AP} {P} name?
			What is the {P} of {S}?
			{PRON} is {S}{AP} {P}?
	Faz parte de algo	{S} is member of the {O}{AP} {P}.	{PRON} {P} is {S} part of?
		{S} is one of the {O} {P}{AP} members.	
		{O} {VHA} {S} as a member of {PRPOSS} {P}.	
	Enumeração		Cite at least one of {S}{AP} {P}.
Temporal	Data	{S}{AP} {P} is {O}.	{PRON} is {S}{AP} {P}?

Marcador	Descrição
S	Propriedade *label* do sujeito da tripla
P	Propriedade *label* do predicado da tripla
PV	Um verbo que pode substituir o *label* do predicado da tripla
O	Propriedade *label* do objeto da tripla
OT	Propriedade *type* do objeto da tripla
PRON	Pronome interrogativo
AP	Apóstrofo
TIME	Representação textual de uma data
VTB	Verbo *to be*
ST	Classe do sujeito
A	Artigo
VHA	Verbo *to have*
PRPOSS	Pronome possessivo

APÊNDICE B – CONJUNTO DE SENTENÇAS AVALIADAS NA PRIMEIRA AVALIAÇÃO

What's Bill Gates' birth place?
What's Bill Gates' birth date?
What's Bill Gates' has abstract?
What's Bill Gates' active years start year?
What's Bill Gates' alma mater?
What's Bill Gates' birth name?
What's Bill Gates' birth year?
What's Bill Gates' board?
How much is Bill Gates' networth ($)?
Who is Bill Gates' parent?
What's Bill Gates' person function?
What's Bill Gates' residence?
Who is Bill Gates' spouse?
What's Bill Gates' title?
What's Bill Gates' 1a?
What's Bill Gates' 1p?
What's Bill Gates' 1y?
What's Bill Gates' 2a?
What's Bill Gates' 2p?
What's Bill Gates' 2y?
What's Barack Obama's birth place?
What's Barack Obama's birth date?
What's Barack Obama's name?
What's Barack Obama's has abstract?
What's Barack Obama's active years end date?
What's Barack Obama's active years start date?
What's Barack Obama's alma mater?
What's Barack Obama's order in office?
What's Barack Obama's residence?
What's Barack Obama's seniority?
Who is Barack Obama's spouse?
What's Barack Obama's birth name?
What's Barack Obama's book?
What's Barack Obama's by?
What's Barack Obama's children?
What's Barack Obama's commons?
What's Barack Obama's congbio?
What's Barack Obama's d?
What's Barack Obama's district?
What's Barack Obama's fec?
What's Neymar's birth place?
What's Neymar's birth date?

What's Neymar's height (cm)?
What's Neymar's has abstract?
What's Neymar's career station?
How much is Neymar's height (μ)?
What's Neymar's number?
What's Neymar's align?
What's Neymar's bg?
What's Neymar's bordercolor?
What's Neymar's Caption?
What's Neymar's fg?
What's Neymar's image size?
What's Neymar's nationalcaps?
What's Neymar's nationalgoals?
What's Neymar's nationalteam?
What's Neymar's nationalyears?
What's Neymar's quote?
What's Neymar's source?
What's Neymar's title?
What's Chicago's area metro (km2)?
What's Chicago's area total (km2)?
What's Chicago's area urban (km2)?
What's Chicago's population density (/sqkm)?
What's Chicago's has abstract?
What's Chicago's area code?
How much is Chicago's area land (m2)?
How much is Chicago's area metro (m2)?
How much is Chicago's area total (m2)?
How much is Chicago's area urban (m2)?
How much is Chicago's area water (m2)?
How much is Chicago's elevation (μ)?
What's Chicago's founding date?
What's Chicago's governing body?
Who is Chicago's leader name?
What's Chicago's leader title?
How much is Chicago's maximum elevation (μ)?
How much is Chicago's minimum elevation (μ)?
What's Chicago's motto?
What's Chicago's percentage of area water?
What's Chicago's population as of?
How much is Chicago's population density (/sqkm)?
What's Chicago's population metro?
What's Chicago's population total?
What's Chicago's total population ranking?
What's Chicago's postal code?
What's Chicago's time zone?

What's Chicago's UTC offset?
What's Chicago's align?
What's Chicago's area code type?
What's Porto Alegre's area (km2)?
What's Porto Alegre's area total (km2)?
What's Porto Alegre's population density (/sqkm)?
What's Porto Alegre's has abstract?
How much is Porto Alegre's area (m2)?
What's Porto Alegre's area code?
How much is Porto Alegre's area total (m2)?
How much is Porto Alegre's elevation (µ)?
What's Porto Alegre's founding date?
Who is Porto Alegre's leader name?
What's Porto Alegre's leader title?
What's Porto Alegre's motto?
How much is Porto Alegre's population density (/sqkm)?
What's Porto Alegre's population metro?
What's Porto Alegre's population total?
What's Porto Alegre's postal code?
What's Porto Alegre's time zone?
What's Porto Alegre's UTC offset?
What's Porto Alegre's Apr high C?
What's Porto Alegre's Apr humidity?
What's Porto Alegre's Apr low C?
What's Porto Alegre's Apr mean C?
What's Porto Alegre's Apr precipitation days?
What's Porto Alegre's Apr precipitation mm?
What's Porto Alegre's Apr record high C?
What's Porto Alegre's Apr record low C?
What's Porto Alegre's Apr sun?
What's Porto Alegre's Aug high C?
What's Porto Alegre's Aug humidity?
What's Porto Alegre's Aug low C?
What's Lion's name?
What's Lion's has abstract?
What's Lion's image?
What's Lion's image2 caption?
What's Lion's image2 width?
What's Lion's image caption?
What's Lion's image width?
What's Lion's range map?
What's Lion's range map2 caption?
What's Lion's range map2 width?
What's Lion's range map width?
What's Lion's status?

What's Lion's status system?
What's Lion's subdivision ranks?
What's Lion's synonyms?
What's Lion's taxon?
What's Lion's trend?
What's Tiger's has abstract?
What's Tiger's align?
What's Tiger's c?
What's Tiger's Caption?
What's Tiger's date?
What's Tiger's direction?
What's Tiger's image?
What's Tiger's image caption?
What's Tiger's p?
What's Tiger's range map?
What's Tiger's range map caption?
What's Tiger's status?
What's Tiger's status system?
What's Tiger's subdivision?
What's Tiger's subdivision ranks?
What's Tiger's synonyms?
What's Tiger's taxon?
What's Tiger's title?
What's Tiger's trend?
What's Tiger's url?
What's Tiger's width?
What's Influenza's has abstract?
What's Influenza's ICD10?
What's Influenza's ICD9?
What's Influenza's MeSH ID?
What's Influenza's OMIM id?
What's Influenza's Caption?
What's Influenza's DiseasesDB?
What's Influenza's eMedicineSubj?
What's Influenza's eMedicineTopic?
What's Influenza's field?
What's Influenza's MedlinePlus?
What's Influenza's width?
vvvv
What's World War II's causalties?
What's World War II's combatant?
What's World War II's place of military conflict?
What's World War II's result?
What's World War II's combatants header?
What's World War II's b?
What's World War II's book?

What's World War II's Caption?
What's World War II's casualties?
What's World War II's commons?
What's World War II's d?
What's World War II's date?
What's World War II's n?
What's World War II's portal?
What's World War II's q?
What's World War II's s?
What's World War II's v?
What's World War II's wikt?
What's French Revolution's has abstract?
What's French Revolution's date?
What's French Revolution's Event Name?
What's French Revolution's image caption?
What's French Revolution's image name?
What's French Revolution's label?
What's French Revolution's location?
What's French Revolution's onlinebooks?
What's French Revolution's participants?
What's French Revolution's result?
What's French Revolution's title?

APÊNDICE C - CONJUNTO DE SENTENÇAS AFIRMATIVAS GERADAS NA SEGUNDA AVALIAÇÃO

Barack Obama is a person whose birth place is Hawaii.
Barack Obama's birth date is on August, 4th 1961.
Barack Obama's institution is Occidental College.
Barack Obama's residence is White House.
Barack Obama's woman is Michelle Obama.
Bill Gates' birth place is Seattle.
Bill Gates' birth date is on [][][]
Bill Gates' attended Harvard University.
Bill Gates is a person whose birth name is William Henry Gates III.
Bill Gates' birth year was 1955.
Bill Gates is an agent whose one is Berkshire Hathaway.
Bill Gates is an agent whose parent is Mary Maxwell Gates.
Bill Gates' {PV} http://dbpedia.org/resource/Bill_Gates__1.
Bill Gates is a person whose residence is Medina, Washington.
Bill Gates' spouse is Melinda Gates
Neymar's birth place is São Paulo.
Neymar's birth date is on February, 5th 1992.
Neymar is a person whose career station is
http://dbpedia.org/resource/Neymar__1
Elvis Presley's birth place is Memphis, Tennessee.
Elvis Presley's death date is on August, 16th 1977.
Elvis Presley's death place is Tupelo, Mississippi.
Elvis Presley's birth date is on January, 8th 1935.
Elvis Presley's birth name is Elvis Aron Presley.
Elvis Presley's birth year was 1935.
Elvis Presley's cub is Lisa Marie Presley.
Elvis Presley's death year was 1977.
Elvis Presley's resting place is Graceland.
Elvis Presley's {PV}
http://dbpedia.org/resource/Elvis_Presley__restingPlacePosition__1.
Elvis Presley is a person whose spouse is Priscilla Presley.
Arnold Schwarzenegger's birth place is Thal, Styria.
Arnold Schwarzenegger's birth date is on July, 30th 1947.
Arnold Schwarzenegger's went to University of Wisconsin–Superior.
Arnold Schwarzenegger's child is Katherine Schwarzenegger.
Arnold Schwarzenegger is a person whose military branch is Austrian Armed Forces.
Arnold Schwarzenegger's president is George H. W. Bush.
Arnold Schwarzenegger's service end year was 1965.
Arnold Schwarzenegger's service start year was 1965.
Arnold Schwarzenegger is an person whose trademark is Arnold Schwarzenegger Signature.svg.
Arnold Schwarzenegger's spouse is Maria Shriver.

Charlie Chaplin's death place is Switzerland.
Charlie Chaplin's death date is on December, 25th 1977.
Charlie Chaplin's birth place is London.
Charlie Chaplin's birth date is on April, 16th 1889.
Charlie Chaplin's birth name is Charles Spencer Chaplin.
Charlie Chaplin's birth year was 1889.
Charlie Chaplin's death year was 1977.
Charlie Chaplin's relative is http://dbpedia.org/resource/Chaplin_family.
Charlie Chaplin's spouse is Oona O'Neill.
Brazil's area total (km2) has 8514837.141576303.
Brazil's population density (/sqkm) has 23.8.
Brazil's area total (m2) has 8.51484e+12.
Brazil's capital is Brasília.
Brazil is a populated place whose largest city is São Paulo.
Brazil is a populated place whose leader title is President.
Brazil's official language is Portuguese language.
Seattle's area (km2) has 1.0E-6.
Seattle's area metro (km2) has 21201.642671210495.
Seattle's area total (km2) has 369.07330572288.
Seattle's population density (/sqkm) has 3150.979715864901.
Seattle's area code is 206.
Seattle's area land (m2) has 2.172e+08.
Seattle's area total (m2) has 3.69073e+08.
Seattle's area water (m2) has 1.51955e+08.
Seattle's governing body is Seattle City Council.
Seattle's leader name is Ed Murray (Washington politician).
Seattle is a city whose leader title is Mayor.
Porto Alegre's area (km2) has 1.0E-6.
Porto Alegre's area total (km2) has 496.827.
Porto Alegre's population density (/sqkm) has 3030.0.
Porto Alegre's area code is +55 51.
Porto Alegre's area total (m2) has 4.96827e+08.
Porto Alegre's elevation (μ) has 10.0.
Porto Alegre's leader name is Democratic Labour Party (Brazil).
Porto Alegre's leader title is Mayor.
Porto Alegre's population density (/sqkm) has 3030.0.
Porto Alegre is a settlement whose time zone is http://dbpedia.org/resource/UTC.
United States' area total (km2) has 9833516.638013326.
United States' population density (/sqkm) has 34.980855563945596.
United States' area total (m2) has 9.83352e+12.
United States is a populated place whose capital is Washington, D.C..
United States' clan is http://dbpedia.org/resource/White_American.
United States is a populated place whose largest city is New York City.
United States is a populated place whose leader title is Chief Justice.
United States' official language is English

APÊNDICE D – CONJUNTO DE SENTENÇAS INTERROGATIVAS GERADAS NA SEGUNDA AVALIAÇÃO

What is the birth place of Barack Obama?
When is Barack Obama's birth date?
What is Barack Obama's academy?
Where is Barack Obama's residence?
Who is Barack Obama's spouse?
Where is Bill Gates' birth place?
When is Bill Gates's birth date?
What is Bill Gates' alma mater?
What is the birth name of Bill Gates?
Which year was Bill Gates' birth year?
What is Bill Gates' mess?
Who is Bill Gates' originator?
What is the person function of Bill Gates?
Where is Bill Gates' residence?
Who is Bill Gates' spouse?
Where is Neymar's birth place?
When is Neymar's birth date?
What is the career station of Neymar?
What is the Elvis Presley's death place name?
What is Elvis Presley's death date?
What is Elvis Presley's birth place?
When is Elvis Presley's birth date?
What is Elvis Presley's birth name?
Which year was Elvis Presley's birth year?
What is the Elvis Presley's child name?
Which year was Elvis Presley's death year?
What is the resting place of Elvis Presley?
What is Elvis Presley's resting place position?
What is the name of Elvis Presley's wife?
What is Arnold Schwarzenegger's birth place?
When is Arnold Schwarzenegger's birth date?
What is Arnold Schwarzenegger's institution?
What is the Arnold Schwarzenegger's child name?
What is the military branch of Arnold Schwarzenegger?
Who is Arnold Schwarzenegger's leader?
Which year was Arnold Schwarzenegger's service end year?
Which year was Arnold Schwarzenegger's service start year?
What is the impression of Arnold Schwarzenegger?

What is the name of Arnold Schwarzenegger's wife?
What is the death place of Charlie Chaplin?
When is Charlie Chaplin's death date?
What is the birth place of Charlie Chaplin?
When is Charlie Chaplin's birth date?
What is Charlie Chaplin's birth name?
Which year was Charlie Chaplin's birth year?
Which year was Charlie Chaplin's death year?
Who did Charlie Chaplin {PV} null?
What is the name of Charlie Chaplin's spouse?
What is Brazil's area total (km2)?
What is Brazil's population density (/sqkm)?
How much is Brazil's area total (m2)?
What is the name of Brazil's capital?
What is the largest city of Brazil?
What is Brazil's leader title?
What is the official language of Brazil?
What is Seattle's area (km2)?
What is Seattle's area metro (km2)?
What is Seattle's area total (km2)?
What is Seattle's area total (km2)?
What is Seattle's area code?
How much is Seattle's area land (m2)?
How much is Seattle's area total (m2)?
How much is Seattle's area water (m2)?
What is Seattle's governing body?
What is the Seattle's commander name?
What is the leader title of Seattle?
What is Porto Alegre's area (km2)?
What is Porto Alegre's area total (km2)?
What is Porto Alegre's population density (/sqkm)?
What is Porto Alegre's area code?
How much is Porto Alegre's area total (m2)?
How much is Porto Alegre's elevation (µ)?
What is the leader name of Porto Alegre?
What is the leader title of Porto Alegre?
How much is Porto Alegre's population density (/sqkm)?
What is Porto Alegre's UTC offset?
What is United States' area total (km2)?
What is United States' population density (/sqkm)?
How much is United States' area total (m2)?
What is the name of United States' capital?
What is the name of United States' ethnic group?

What is the United States' largest city name?
What is the United states' leader title?
What is United States' official language?

APÊNDICE E – CONJUNTO DE SENTENÇAS AFIRMATIVAS DA TERCEIRA AVALIAÇÃO

Barack Obama, whose birth place is Hawaii - Honolulu, is a politician.
Barack Obama's birth date is on August, 4th 1961.
Barack Obama's residence is White House.
Barack Obama's colleges are Occidental College, Columbia College, Columbia University and Harvard Law School.
Barack Obama's wife is Michelle Obama.
Hillary Clinton is an office holder whose birth place is Chicago.
Hillary Clinton's birth date is on October, 26th 1947.
Hillary Clinton's colleges are Yale University and Wellesley College.
Hillary Clinton's child is Chelsea Clinton.
Republican Party (United States) has Hillary Clinton as a member of their other party.
Hillary Clinton's presidents are Bill Clinton and Barack Obama.
Hillary Clinton's partner is Bill Clinton.
Bob Dylan's birth place is Duluth, Minnesota.
Bob Dylan's birth date is on May, 24th 1941.
Bob Dylan's birth name is Robert Allen Zimmerman.
Bob Dylan's birth year was 1941.
Bob Dylan's kids are Jakob Dylan and Jesse Dylan.
Bob Dylan's home town is Hibbing, Minnesota.
Bob Dylan's residence is Malibu, California.
Bob Dylan's mates are Carolyn Dennis and Sara Dylan.
Michael Jackson's death place is Los Angeles.
Michael Jackson, whose birth place is Gary - Indiana, is an artist.
Michael Jackson's death date is on June, 25th 2009.
Michael Jackson's birth date is on August, 29th 1958.
Ben Affleck's birth places are Berkeley, California.
Ben Affleck's birth date is on August, 15th 1972.
Ben Affleck's colleges are Occidental College and University of Vermont.
Ben Affleck's birth name is Benjamin Geza Affleck-Boldt.
Ben Affleck's birth year was 1972.
Ben Affleck's relation is Casey Affleck.
Ben Affleck's wife is Jennifer Garner.
Russia has a population density of 8.4
Russia has a area total (m2)S of 1.70752e+13
Russia's capital is Moscow.
Russia's ethnic groups are Tatars, Ukrainians in Russia, Chuvash people and Bashkirs.
Russia's leader titles are President, Chairman of the State Duma, Prime Minister and Chairman of the Federation Council.
Russia's official language is Russian language.
Canada has a population density of 3.41.

Canada is a country whose ethnicities are Aboriginal peoples in Canada and European Canadian.
Canada's largest city is Toronto.
Canada's leader titles are Chief Justice, Monarch, Governor General and Prime Minister.
Canada, whose official languages are French language and English language, is a country.
Canada, whose regional languages are Cree language and Dogrib language, is a country.
Canada's area total (m2)S is 9.98203e+12.
Canada's section is Ottawa.
United States have a population densitity of 35.
United States' area total (m2) is 9.83352e+12.
United States' largest city is New York City.
United States' leader titles are Chief Justice, President, Vice President and Speaker of the House.
United States, whose official language is Federal government of the United States, is a country.
United States, whose capital is Washington - D.C., is a person.
Virginia has a area total (km2) of 110785.67
Virginia, whose capital is Richmond, Virginia, is a place.
Virginia's largest city is Virginia Beach, Virginia.
Virginia's maximum elevation (Î¼) is 1746.0.
Virginia's minimum elevation (Î¼) is 0.0.
Chicago has a area metro (km2) of 28163.530711793665
Chicago has a area total (km2) of 606.057217818624
Chicago's area urban (km2) is 5498.026760621261.
Chicago has a population density of 4447.4.
Chicago is a place whose are code is 312/872and773/872.
Chicago's area land (m2) is 5.88704e+08.
Chicago has a elevation (Î¼) of 181.051
Chicago's government body is Chicago City Council.
Chicago is a city whose leader names are Susana Mendoza, Rahm Emanuel and Kurt Summers, Jr..
Chicago, whose leader titles are City Clerk, Mayor and City Treasurer, is a city.
Chicago's maximum elevation (Î¼) has 204.826.
Chicago's minimum elevation (Î¼) has 176.174.
Porto Alegre has a area (km2) of 1.0E-6
Porto Alegre has a population density of 3030.0.
Porto Alegre's area code is +55 51.
Porto Alegre's area total (m2) has 4.96827e+08.
Porto Alegre's elevation (Î¼) has 10.0.
Porto Alegre is a place whose leader name is Democratic Labour Party (Brazil).
Porto Alegre's leader title is Mayor.
Porto Alegre, whose UTC offset are -2 and -3, is a place.

APÊNDICE F – CONJUNTO DE SENTENÇAS INTERROGATIVAS DA TERCEIRA AVALIAÇÃO

Where is Barack Obama's birth place?
When is Barack Obama's birth date?
Cite at least one of Barack Obama's colleges.
Where is Barack Obama's residence?
What is the Barack Obama's wife name?
What is the birth place of Bob Dylan?
When is Bob Dylan's birth date?
What is the birth name of Bob Dylan?
Which year was Bob Dylan's birth?
Cite at least one of Bob Dylan's sisters.
What is the home town of Bob Dylan?
What is the name of Bob Dylan's residence?
Cite at least one of Bob Dylan's pairs.
What is the name of Hillary Clinton's birth place?
When is Hillary Clinton's birth date?
Cite at least one of Hillary Clinton's colleges.
Who is Hillary Clinton's child?
What other party is Hillary Clinton part of?
Cite at least one of Hillary Clinton's president.
Who is Hillary Clinton's husband?
Cite at least one of Ben Affleck's birth places.
When is Ben Affleck's birth date?
Cite at least one of Ben Affleck's colleges.
Which year was Ben Affleck's birth?
What is Ben Affleck's birth name?
What is the Ben Affleck's relation name?
Who is Ben Affleck's partner?
What is the death place of Michael Jackson?
When is Michael Jackson's death date?
What is the Michael Jackson's birth place name?
When is Michael Jackson's birth date?
What is Russia's population density?
Cite at least one of Russia's ethnic groups.
What is Russia's leader titles?
What is Russia's official language?
How much is Russia's area total (m2)S?
What is the capital of Russia?
What is Canada's population density?
How much is Canada's area total (m2)S?
What is the Canada's capital name?
Cite at least one of Canada's ethnics.
What is the name of Canada's largest city?
What is Canada's leader titles?
Cite at least one of Canada's official languages.

Cite at least one of Canada's regional languages.
What is United States' population density?
How much is United States' area total (m2)?
What is United States' official language?
What is the largest city of United States?
What is the leader titles of United States?
What is the United States' capital name?
How much is Virginia's area total (m2)?
What is the Virginia's capital name?
What is the name of Virginia's largest city?
How much is Virginia's maximum elevation (Î¼)?
How much is Virginia's minimum elevation (Î¼)?
What is Porto Alegre's area total (km2)?
What is Porto Alegre's area (km2)?
What is Porto Alegre's population density?
What is Porto Alegre's cartesian product?
How much is Porto Alegre's elevation (Î¼)?
What is the leader name of Porto Alegre?
What is Porto Alegre's leader title?
What is Porto Alegre's UTC offset?
What is Chicago's area metro (km2)?
What is Chicago's area total (km2)?
What is Chicago's area urban (km2)?
What is Chicago's population density?
What is the area code of Chicago?
How much is Chicago's area land (m2)?
Cite at least one of Chicago's leader names.
What is the leader titles of Chicago?
How much is Chicago's maximum elevation (Î¼)?
How much is Chicago's minimum elevation (Î¼)?
How much is Chicago's elevation (Î¼)?
What is the governing body of Chicago?