Universidade Federal de Campina Grande
Centro de Engenharia Elétrica e Informática
Coordenação de Pós-Graduação em Ciência da Computação

Dissertação de Mestrado

Um Modelo BERT para Sumarização Extrativa de Textos em Documentos da Polícia Federal

Thierry Silva Barros

Campina Grande, Paraíba, Brasil

04/2022

Universidade Federal de Campina Grande
Centro de Engenharia Elétrica e Informática
Coordenação de Pós-Graduação em Ciência da Computação

Um Modelo BERT para Sumarização Extrativa de Textos em Documentos da Polícia Federal

Thierry Silva Barros

Dissertação submetida à Coordenação do Curso de Pós-Graduação em Ciência da Computação da Universidade Federal de Campina Grande - Campus I como parte dos requisitos necessários para obtenção do grau de Mestre em Ciência da Computação.

Área de Concentração: Ciência da Computação
Linha de Pesquisa: Sistemas da Informação

Carlos Eduardo Santos Pires (UFCG)
Dimas Cassimiro do Nascimento Filho (UFAPE)
(Orientadores)

Campina Grande, Paraíba, Brasil

B277m Barros, Thierry Silva.
Um modelo BERT para sumarização extrativa de textos em documentos da Polícia Federal / Thierry Silva Barros. – Campina Grande, 2022.
89 f.

Dissertação (Mestrado em Ciência da Computação) – Universidade Federal de Campina Grande, Centro de Engenharia Elétrica e Informática, 2022.
"Orientação: Prof. Dr. Carlos Eduardo Santos Pires, Dimas Cassimiro do Nascimento Filho"
Referências.

1. Processamento de Linguagem Natural. 2. BERT. 3. Notícia-crime. 4. Sumarização Automática de Texto. 5. Polícia Federal do Brasil. 6. Investigação Policial. I. Pires, Carlos Eduardo Santos. II. Nascimento Filho, Dimas Cassimiro do. III. Título.

CDU 004.438:81’322.2(043)

FICHA CATALOGRÁFICA ELABORADA PELA BIBLIOTECÁRIA MARIA ANTONIA DE SOUSA CRB 15/398

MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
POS-GRADUACAO CIENCIAS DA COMPUTACAO
Rua Aprigio Veloso, 882, - Bairro Universitario, Campina Grande/PB, CEP 58429-900

FOLHA DE ASSINATURA PARA TESES E DISSERTAÇÕES

THIERRY SILVA BARROS

UM MODELO BERT PARA SUMARIZAÇÃO EXTRATIVA DE TEXTOS EM DOCUMENTOS DA POLÍCIA FEDERAL

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação como pré-requisito para obtenção do título de Mestre em Ciência da Computação.

Aprovada em: 28/04/2022

Prof. Dr. CARLOS EDUARDO SANTOS PIRES,, UFCG, Orientador

Prof. Dr. DIMAS CASSIMIRO DO NASCIMENTO FILHO, UFRPE, Orientador

Prof. Dr. LEANDRO BALBY MARINHO, UFCG, Examinador Interno

Prof. Dr. FREDERICO LUIZ GONÇALVES DE FREITAS, UFPE, Examinador Externo

Documento assinado eletronicamente por **CARLOS EDUARDO SANTOS PIRES**, **PROFESSOR 3 GRAU**, em 28/04/2022, às 16:07, conforme horário oficial de Brasília, com fundamento no art. 8º, caput, da Portaria SEI nº 002, de 25 de outubro de 2018.

Documento assinado eletronicamente por **Dimas Cassimiro do Nascimento Filho**, **Usuário Externo**, em 28/04/2022, às 16:56, conforme horário oficial de Brasília, com fundamento no art. 8º, caput, da Portaria SEI nº 002, de 25 de outubro de 2018.

Documento assinado eletronicamente por **LEANDRO BALBY MARINHO**, **PROFESSOR DO MAGISTERIO SUPERIOR**, em 29/04/2022, às 14:08, conforme horário oficial de Brasília, com fundamento no art. 8º, caput, da Portaria SEI nº 002, de 25 de outubro de 2018.

A autenticidade deste documento pode ser conferida no site https://sei.ufcg.edu.br/autenticidade, informando o código verificador **2318751** e o código CRC **E68AB5FE**.

Referência: Processo nº 23096.023218/2022-30 SEI nº 2318751

Resumo

Na Polícia Federal do Brasil, um documento denominado notícia-crime é utilizado como ponto de partida em qualquer investigação criminal. Uma notícia-crime tem como objetivo fornecer um resumo das atividades investigativas e, para tal, deve conter todas as informações relevantes sobre o suposto crime ocorrido. A fim de administrar uma investigação e correlacionar com investigações semelhantes, em geral, a Polícia Federal precisa extrair as informações mais importantes do documento da notícia-crime. A extração manual (ler e compreender todo o seu conteúdo) tende a ser exaustiva, devido ao tamanho e à complexidade dos documentos. Neste sentido, técnicas de Processamento de Linguagem Natural (PLN) podem auxiliar na extração automática dos trechos mais importantes como, por exemplo, o crime ocorrido. Nos últimos anos, as redes neurais profundas têm sido aplicadas com sucesso em muitas tarefas diferentes de PLN. Um modelo de rede neural que alavancou os resultados em uma ampla gama de tarefas de PLN foi o modelo BERT (*Bidirectional Encoder Representations from Transformers*). Devido à sua capacidade de representação do sentido de dados textuais, o modelo consegue capturar dependências de curto (correlações entra dados textuais que estão próximos no texto) e longo (correlações entra dados textuais que estão distantes no texto) alcance nos dados textuais. O presente trabalho propõe diferentes abordagens baseadas no modelo BERT para extrair as informações mais importantes do documento textual referente a uma notícia-crime e construir um resumo do mesmo. Para a sumarização automática de documentos textuais podem ser aplicados dois tipos de técnicas diferentes: abstrativa e extrativa. Nesta pesquisa foi utilizada nas abordagens a técnica de sumarização extrativa para resumo dos documentos. A viabilidade da utilização do modelo BERT para extrair e sintetizar as informações mais importantes de uma notícia-crime é avaliada em termos de eficácia e eficiência. Para tal, são utilizados dois conjuntos de dados reais: o conjunto de dados da Polícia Federal (de domínio privado) e o conjunto de dados Wikihow brasileiro (de domínio público). Os resultados experimentais, usando diferentes variantes da métrica ROUGE, mostram que as abordagens propostas podem aumentar significativamente a eficácia do resumo de texto extrativo sem sacrificar a eficiência.

Palavras-chave: processamento de linguagem natural, BERT, notícia-crime, sumarização automática de texto, Polícia Federal do Brasil, investigação policial.

Abstract

In the Federal Police, a document known as *notitia criminis* is used as the starting point of the criminal investigation. The *notitia criminis* document aims to report a summary of investigative activities and contains all relevant information about the supposed crime that occurred. In order to manage an investigation and correlate with similar investigations, in general, the Federal Police needs to extract the most important information of the *notitia criminis* document. Manual extraction (reading and understand their entire content) may be human exhausting, due to the size and complexity of the documents. Therefore, it is necessary to use Natural Language Processing (NLP) techniques for automatically extracting the most important passages, such as the crime that occurred.In the last few years, deep neural networks have been successfully applied to many different NLP tasks. A neural network model that leveraged the results in a wide range of NLP tasks was the BERT model - an acronym for Bidirectional Encoder Representations from Transformers. Due to its ability to represent the meaning textual data, being able to capture both short-range (correlations between textual data that are close together in the text) and long-range (correlations between textual data that are far apart in the text) dependence on textual data. This dissertation proposes different approaches based on the BERT model to extract the most important information from the textual document referring to a *notitia criminis* document and build a summary of it. For the automatic summarization of textual documents, two types of different techniques can be applied: abstractive and extractive. In this dissertation, the extractive summarization technique was used to summarize the documents. Thus, we aim to analyze the feasibility of using the BERT model to extract and synthesize the most important information from the *notitia criminis* document. We evaluate the performance of the proposed approaches using two real datasets: the Federal Police dataset (a private domain dataset) and the Brazilian Wikihow dataset (a public domain dataset). Experimental results on the two datasets, using different variants of the ROUGE metric, show that our approaches can significantly increase extractive text summarization effectiveness without sacrificing efficiency.

Keywords: natural language processing, BERT, *notitia criminis*, automatic text summarization, Brazilian Federal Police, police investigation.

Agradecimentos

A Deus, todas as minhas realizações. Em primeiro lugar a ele que é minha base e fortaleza, minha luz a seguir. A fé e persistência me permitiram superar muitas dificuldades e limitações nos momentos mais difíceis.

À minha família, principalmente meus pais, que me ensinaram os primeiros caminhos da educação e me permitiram seguir os meus sonhos, da forma deles. Aos meus irmão que sempre estiveram ao meu lado, me dando forças e motivações, sempre me incentivaram a manter o foco nos estudos e me ajudaram na caminhada até aqui.

Aos meus orientadores, Carlos Eduardo e Dimas Cassimiro, que são exemplos de seres humanos e ótimos profissionais, sempre demonstram muita dedicação e responsabilidade com suas atividades, além de me incentivarem e me ensinarem princípios que vou levar para o resto da vida. Gratidão por todo incentivo, apoio e paciência neste período que passamos juntos.

Aos meus colegas do Laboratório de Práticas de Software (SPLab). Aos desenvolvedores e professores que sempre me incentivaram e me ajudaram a realizar as atividades da pesquisa, apresentando sugestões, alternativas e, principalmente, pelos momentos de diversão.

A todos os meus colegas da UFCG, pelas contribuições diretas e indiretas nesta pesquisa, bem como os muitos momentos de descontração e trocas de conhecimento.

Ao projeto ePol por todo o incentivo e o suporte a pesquisa.

Aos professores da COPIN e todos que um dia tive a honra de ser aluna, desde a infância. Tenho muita gratidão e admiração pelos mestres educadores.

Conteúdo

1 Introdução **1**

1.1 Motivação 4

1.2 Relevância 6

1.3 Objetivos 6

1.4 Contribuições 7

1.5 Organização do trabalho 8

2 Fundamentação Teórica **10**

2.1 Conceitos de Aprendizagem de Máquina 10

2.1.1 Definição de Aprendizagem de Máquina 10

2.1.2 Abordagem *Ensemble* 13

2.2 Conceitos de Processamento de Linguagem Natural 15

2.2.1 Análise Sintática e Semântica de Documentos Textuais 15

2.2.2 Pré-processamento de Documentos Textuais 16

2.2.3 Sumarização Automática de Documentos Textuais 16

2.2.4 Sumarização Extrativa 17

2.2.5 Sumarização Abstrativa 18

2.2.6 Modelos pré-treinados 18

2.2.7 Métricas para Avaliação da Qualidade de Resumos 21

2.3 Considerações Finais 23

3 Trabalhos Relacionados **27**

3.1 Metodologia 27

3.2 SA de Texto 28

3.3 Considerações Finais 32

4 Abordagens BERT para Sumarização de Documentos Textuais 35

4.1 Formalização do Problema de Sumarização Extrativa 35

4.1.1 Definição de SA de texto 36

4.2 Características dos Documentos das Notícias Crime 36

4.3 Abordagens para Sumarização Automática de Documentos Textuais 39

4.3.1 BERTSUM-ALD 43

4.3.2 BERTSUM-ALD-MD 45

4.3.3 BERTSUM-ALD-ES 49

4.3.4 Etapa de Treinamento 52

4.3.5 Etapa de Teste 54

4.4 Considerações Finais 54

5 Avaliação Experimental 56

5.1 Questões de Pesquisa 57

5.2 Coleta de Dados 57

5.3 Métricas Utilizadas 59

5.4 Testes Estatísticos 59

5.4.1 Teste de Mann-Whitney 60

5.4.2 Teste de Friedman 60

5.5 Ajuste dos Hiperparâmetros 60

5.6 Bases de Dados 61

5.6.1 Base de Dados de NCs 61

5.6.2 Base de Dados da Wikihow 61

5.7 Etapas de pré-processamento 62

5.7.1 Limpeza do Dados 62

5.7.2 Preparação dos Dados 62

5.7.3 Rotulação dos Dados 62

5.7.4 Preparação dos dados para o modelo BERT 63

5.7.5 Particionamento dos Conjuntos de Dados 63

5.8 Experimentos 64

5.8.1 Avaliação da Eficácia 64
5.8.2 Avaliação da Eficiência 74
5.9 Discussão dos Resultados 75
5.10 Ameaças à Validade 79
5.11 Considerações Finais 79

6 Conclusões e Trabalhos Futuros **80**
6.1 Conclusões 80
6.2 Trabalhos Futuros 82

A Parâmetros dos Modelos **93**

B Tempo de Execução dos Experimentos **97**

Lista de Símbolos

AM - *Apredizagem de Máquina*

Bagging - *Bootstrap Aggregation*

BERT - *Bidirectional Encoder Representations from Transformers*

BERTSUM-ALD - *Abordagem de Sumarização BERT para Documentos de Tamanho Arbitário*

BERTSUM-ALD-ES - *Abordagem de Sumarização BERT com Ensemble para Documentos de Tamanho Arbitário*

BERTSUM-ALD-MD - *Abordagem de Sumarização BERT para Documentos de Tamanho Arbitário e Diferentes Domínios*

BoW - *Bag of Words*

CLS - *token de classificação*

F1 - $F_{measure}$

IA - *Inteligência Artificial*

LCS - *Longest Common Subsequence*

LSTM - *Long short-term memory*

NC - *Notícia-crime*

PLN - *Processamento de Linguagem Natural*

QP - *Questão de Pesquisa*

RNR - *Rede Neural Recorrente*

ROUGE - *Recall-Oriented Understudy for Gisting Evaluation*

ROUGE-L - *Subsequência comum mais longa*

ROUGE-1 - *Unigrama*

ROUGE-2 - *Bigrama*

ROUGE-3 - *Trigrama*

SEP - *token de separação*

SPLab - *Laboratório de Práticas de Software*

TF-IDF - *Term Frequency - Inverse Document Frequency*

Lista de Figuras

1.1 Exemplo do fluxo de instauração de um inquérito policial. 2

1.2 Exemplo de um documento de NC. Um resumo extrativo do documento é feito extraindo e concatenando os textos de tarja amarela. Textos com listras pretas são informações privadas e por isso foram omitidos. 5

2.1 Funcionamento do algortimo de agrupamento *K-Means*. 13

2.2 Exemplo de *ensemble* com abordagem *Bagging* 15

2.3 Comparação entre a sumarização extrativa e abstrativa 17

2.4 Fluxo do processo de sumarização extrativa 18

2.5 Fluxo do processo de sumarização abstrativa 19

2.6 Arquitetura *Transformer* 24

2.7 Predição de Linguagem Mascarada e Predição da Próxima Frase 25

2.8 Arquitetura BERT: BERTSUM vs. BERT 26

2.9 Exemplo da métrica ROUGE-1 26

4.1 Densidade do número de sentenças nos documentos das NCs. 38

4.2 Densidade do número de páginas nos documentos das NCs. 39

4.3 Densidade das posições das sentenças mais importantes nos documentos das NCs. 40

4.4 Áreas de atribuição dos documentos das NCs. 41

4.5 Formulário e arquivo extraídos de um documento de NC. 42

4.6 Abordagem BERTSUM para SA de documentos de tamanho arbitrário . . . 43

4.7 Visualização de como as abordagens lidam com documentos arbitrariamente longos. 45

4.8 Exemplo da abordagem BERTSUM para sumarização de documentos textuais de tamanho arbitário e multidomínios 46
4.9 O fluxo de execução utilizando a abordagem de sumarização automática de texto multidomínio. 47
4.10 Exemplo da abordagem BERTSUM-ALD-ES para sumarização automática de documentos textuais 51

5.1 Um exemplo do nosso conjunto de dados da Wikihow: conjunto de dados da WikiHow, que inclui mais de 100 mil documentos. 58
5.2 Avaliação da eficácia da abordagem BERTSUM-ALD nos conjuntos de dados NC e Wikihow 66
5.3 Avaliação da precisão, cobertura e $F_{measure}$ da abordagem BERTSUM-ALD no conjunto de dados de NCs 67
5.4 Avaliação da precisão, cobertura e $F_{measure}$ da abordagem BERTSUM-ALD no conjunto de dados da Wikihow 68
5.5 Avaliação da eficácia da abordagem BERTSUM-ALD-MD nos conjuntos de dados NC e Wikihow 69
5.6 Avaliação da precisão, cobertura e $F_{measure}$ da abordagem BERTSUM-ALMD no conjunto de dados de NCs 70
5.7 Avaliação da precisão, cobertura e $F_{measure}$ da abordagem BERTSUM-ALD-MD no conjunto de dados da Wikihow 71
5.8 Avaliação da eficácia da abordagem BERTSUM-ALD-ES nos conjuntos de dados 72
5.9 Avaliação da precisão, cobertura e $F_{measure}$ da abordagem BERTSUM-ALES no conjunto de dados de NCs 73
5.10 Avaliação da precisão, cobertura e $F_{measure}$ da abordagem BERTSUM-ALD-ES no conjunto de dados da Wikihow 74
5.11 Comparação da precisão, cobertura e $F_{measure}$, em termos de ROUGE-1, das abordagens propostas com os modelos *baselines*, no conjunto de dados da Wikihow 75

5.12 Comparação da precisão, cobertura e $F_{measure}$, em termos de ROUGE-1, das abordagens propostas com os modelos *baselines*, no conjunto de dados de NCs 76

5.13 Densidade dos comprimentos dos sumários gerados pelos modelos no conjunto de dados de NCs 77

5.14 Tempo de execução, por abordagem, para a predição das sentenças mais importantes dos documentos. 78

6.1 Fluxo de execução da aprendizagem incremental. 84

Lista de Tabelas

3.1 Resumo dos trabalhos relacionados à sumarização automática de texto . . . 34

5.1 Eficácia dos modelos no conjunto de dados de NCs 71
5.2 Eficácia dos modelos no conjunto de dados da Wikihow 72

A.1 Valores dos parâmetros da abordagem BERTSUM-ALD 94
A.2 Valores dos parâmetros da abordagem BERTSUM-ALD-ES 95
A.3 Valores dos parâmetros da abordagem BERTSUM-ALD-MD 96

B.1 Tempo de treinamento das abordagens propostas. 98
B.2 Tempo de experimentação das abordagens propostas. 98

Lista de Códigos Fonte

4.1 Passo à passo da geração do sumário utilizando a abordagem de BERTSUM-ALD-MD 48
4.2 Passo à passo do treinamento da abordagem de BERTSUM-ALD-ES 50

Capítulo 1

Introdução

No Brasil, cerca de 70 mil inquéritos de competência da Justiça Federal são instaurados por ano [45]. Esses inquéritos são de responsabilidade da Polícia Federal e representam investigações de crimes ocorridos no território brasileiro. Quando uma infração penal ocorre, é necessário que as autoridades competentes sejam notificadas para que possam ser tomadas as ações cabíveis para solucionar o crime. Na Polícia Federal, a notificação do crime pode chegar ou ser gerada de diferentes formas: a) por e-mail; b) em documento de papel; ou c) quando uma pessoa vai até uma delegacia (ou superintendência) e narra um fato criminoso (depoimento). Como consequência, é realizada uma avaliação preliminar para verificar se a notificação representa de fato um crime e, caso seja, é avaliado se é um crime de competência da Polícia Federal. Se confirmado, é aberto um caso para investigar o ocorrido.

A materialização da notificação do crime ocorrido recebe o nome de notícia-crime (NC). Porém, antes de ser aberta uma investigação para apurar o caso, é necessário extrair as informações mais importantes da NC a fim de avaliar se ela representa de fato um crime. Além disso, devem ser procurados casos similares que foram instaurados. Se existirem casos similares à NC em questão, o novo caso que vai ser gerado deve ser encaminhado para o mesmo agente ou unidade responsável pelos casos similares. Por exemplo, se for um crime cibernético o caso deve ser enviado para uma delegacia especializada em serviços de repressão a crimes cibernéticos. Além disso, se for constatado que um caso idêntico já foi instaurado, a NC não gera um novo caso para que não ocorra duplicidade de investigação. Por fim, após o encaminhamento para uma delegacia especializada, é gerado um inquérito policial para investigar o caso.

Na Figura 1.1, é mostrado um exemplo de instauração de um inquérito policial. Após a ocorrência de um crime, o mesmo é notificado através de um depoimento na delegacia. Em seguida, são extraídas as informações mais importantes do depoimento e é analisado se o contéudo do depoimento representa um crime. Em caso afirmativo, é instaurado um inquérito policial a fim de investigar o crime. Por outro lado, se não for constatado crime ou houver uma duplicidade de caso, não é instaurado um inquérito para investigar o caso.

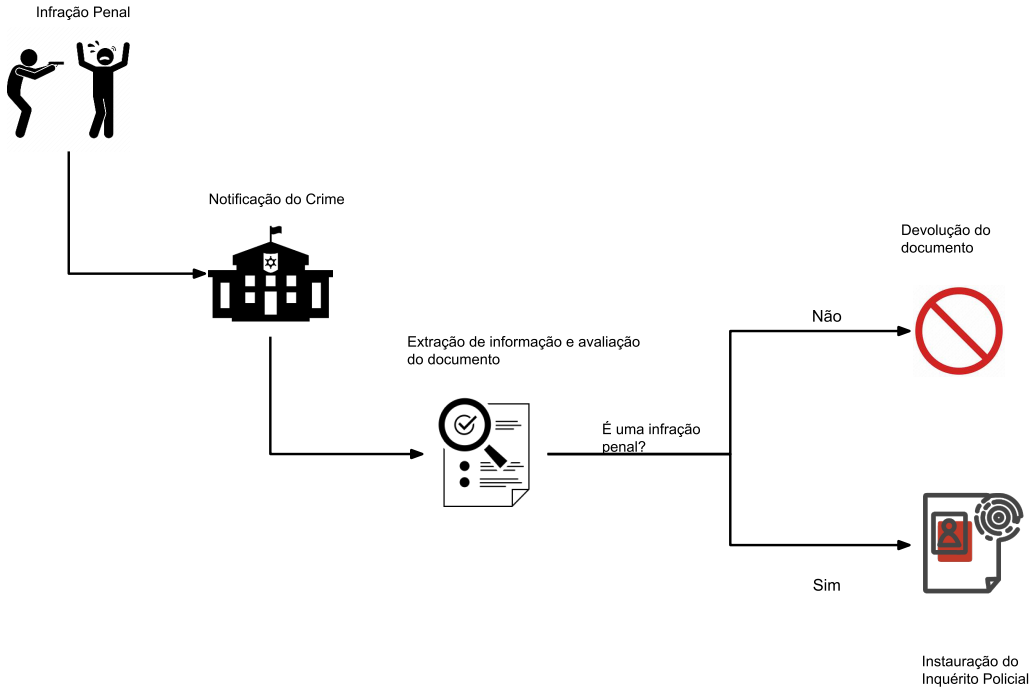


Figura 1.1: Exemplo do fluxo de instauração de um inquérito policial.

No início de um inquérito policial, o documento de NC é comumente utilizado para regular o desenvolvimento de uma investigação [52]. A NC é um procedimento administrativo e representa o ponto de partida do sistema de persecução penal. Considerando o processo penal brasileiro, seu objetivo é denunciar uma série de atividades investigativas. Além disso, é um instrumento formal de investigação policial. A estrutura de um documento de NC pode se diversificar tanto em extensão (de uma a mais de 100 páginas), quanto em sua variação do estilo de escrita que, dependendo do órgão emissor e de qual área de atribuição (subdomínio) faz parte, a forma como foi redigido pode mudar, pois não há um padrão de escrita. Além disso, a complexidade do documento de NC também se deve ao fato de conter todas as informações relevantes sobre o suposto crime ocorrido, ou seja, pode conter várias seções

com provas, depoimentos, imagens, tabelas, etc. Por fim, outra caractéristica das NCs é o formato dos documentos que são em PDF.

Durante uma nova investigação policial, o policial pode consultar um ou mais documentos de NC relacionados para auxiliar na investigação. Devido à estrutura complexa das NCs, torna-se um trabalho exaustivo para um ser humano ler e compreender todo o conteúdo de um documento de NC. A incompreensão da informação presente no documento de NC (por exemplo, se a informação presente constitui um crime real ou qual é a área de atribuição) ou o não reconhecimento de crimes semelhantes pode causar perdas consideráveis em termos de recursos humanos e monetários, pois a má interpretação pode afetar negativamente a investigação do caso. Além disso, milhares de documentos chegam à Polícia Federal todos os dias, tornando essa tarefa custosa para ser realizada por humanos. Portanto, gerar modelos computacionais que resumam de forma eficaz os documentos de NC com as informações mais importantes pode auxiliar o policial a compreender o conteúdo de um ou mais documentos e, ainda, correlacionar casos semelhantes, o que pode impactar a investigação do caso atual.

No entanto, a geração de modelos computacionais para sumarizar NCs enfrenta outro desafio: a grande variedade de subdomínios presente no domínio das NCs da Polícia Federal, o que dificulta a criação de modelos que processem e extraiam informações dos documentos de forma eficaz. São exemplos de subdomínios: divisão de crimes de aplicação da lei, crimes previdenciários, crimes cibernéticos e crimes de desvio de recursos públicos. Assim, um modelo computacional pode funcionar bem para um subdomínio específico e não apresentar o mesmo resultado em outros subdomínios.

Finalmente, gerar um resumo contendo as informações mais importantes de um documento do conjunto de dados de NCs é uma tarefa mais difícil, em termos de eficácia, do que gerar resumos de documentos de outros conjuntos de dados (i.e., conjunto de dados de notícias gerais, esportes), pois as informações importantes podem vir de várias partes do documento e não apenas das primeiras sentenças, o que exige um processamento completo do documento. Além disso, a tarefa de processar e gerar resumos contendo as informações importantes pode ser dificultada pela presença de dados incorretos (e.g. palavras com erros de ortografia) na extração dos textos dos documentos em formato PDF.

1.1 Motivação

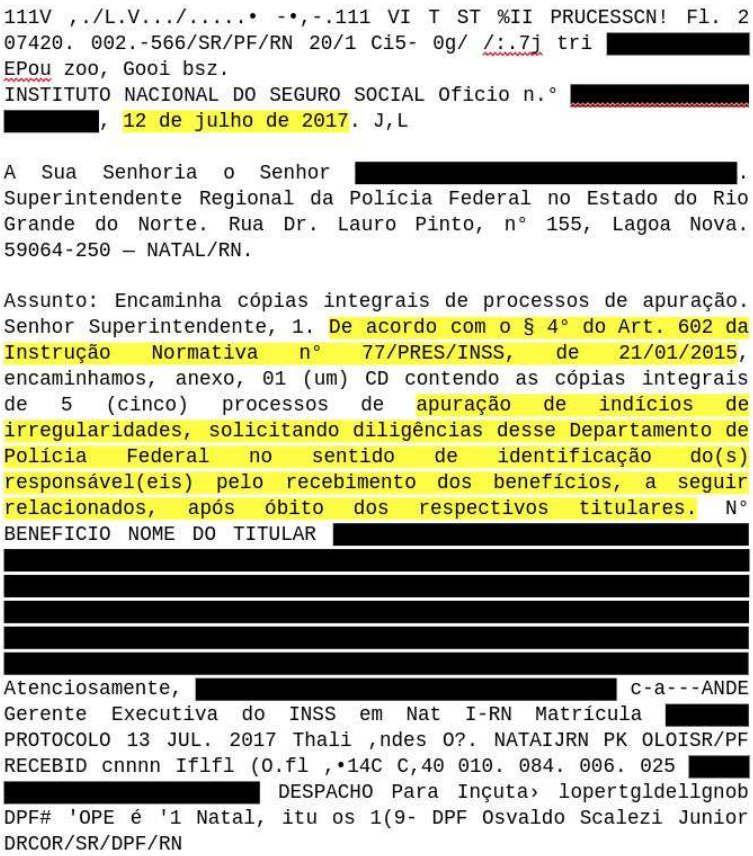
Uma solução para gerar resumos contendo as informações mais importantes de documentos é utilizar uma técnica de Processamento de Linguagem Natural (PLN) conhecida como sumarização automática (SA) de textos [44]. A técnica permite lidar com os documentos das NCs para recuperar as informações mais importantes [23]. A SA de documentos textuais é uma tarefa que visa gerar uma versão mais curta de um documento, mantendo suas informações mais importantes [2]. Para a SA de documentos textuais podem ser aplicados dois tipos de técnicas diferentes: abstrativa e extrativa [66]. Na técnica extrativa, a ideia principal é utilizar as estruturas originais (parágrafos ou sentenças do texto) para gerar uma sumarização, considerando apenas o conteúdo original. A técnica extrativa funciona como um sistema de predição das partes mais importantes do documento. Para ilustrar a tarefa de sumarização extrativa de texto, considere o exemplo de sumarização extrativa em um documento de NC apresentado na Figura 1.2, onde os textos em tarja amarela são extraídos e concatenados, na ordem em que estão presentes no texto, para gerar o resumo.

A técnica abstrativa é mais parecida com a sumarização humana, a qual busca reescrever o texto original de forma reduzida, mantendo os pontos mais relevantes. Esta tarefa requer modelos complexos que dependem de métodos linguísticos para ter uma compreensão profunda do conteúdo do texto. Tais técnicas ainda não apresentam bons resultados em documentos grandes ou em documentos que não são escritos em inglês [62; 46; 71]. Além disso, não foram realizadas pesquisas de sumarização abstrativa de documentos textuais em português, ou seja, não há evidências de que essa técnica possa funcionar bem para documentos nesse idioma. Devido à limitação das técnicas abstrativas em gerar bons resumos para documentos grandes, e à falta de evidências de que esta técnica funciona em documentos na língua portuguesa, este trabalho assumiu que a técnica mais promissora seria a técnica extrativa. Além disso, a facilidade de implementação de técnicas extrativas e o fato de a maioria dos trabalhos existentes [47] seguirem técnicas extrativas fazem com que essa técnica se adeque melhor a diferentes contextos. Portanto, a técnica extrativa foi escolhida para ser utilizada nesta pesquisa. Deste ponto em diante, quando nos referirmos a uma técnica de sumarização de documentos, estaremos nos referindo à técnica extrativa.

Para lidar com o problema de geração de resumos contendo as informações mais im-

portantes dos documentos das NCs, uma predição efetiva das partes mais importantes do documento pode facilitar o entendimento dos documentos por partes dos agentes policiais e, consequentemente, facilitar a tomada de decisões e minimizar as perdas em termos de recursos monétários. No entanto, predizer as partes mais importantes de um documento de NC é um desafio devido as características do documento citadas anteriormente: tamanho do documento e presença de subdomínios.

Dessa forma, foi identificada como possibilidade de pesquisa científica a proposição de uma ou mais abordagens de sumarização extrativa para predição das partes mais importantes de documentos de NC utilizando modelos de SA de texto. Idealmente, tais abordagens devem maximizar a eficácia das predições, ou seja, predizer com precisão as sentenças mais importantes dos documentos.



111V ,./L.V.../.....• -•,-.111 VI T ST %II PRUCESSCN! Fl. 2
07420. 002.-566/SR/PF/RN 20/1 Ci5- 0g/ /:.7j tri [redacted]
EPou zoo, Gooi bsz.
INSTITUTO NACIONAL DO SEGURO SOCIAL Oficio n.° [redacted]
[redacted], 12 de julho de 2017. J,L

A Sua Senhoria o Senhor [redacted].
Superintendente Regional da Polícia Federal no Estado do Rio Grande do Norte. Rua Dr. Lauro Pinto, n° 155, Lagoa Nova. 59064-250 — NATAL/RN.

Assunto: Encaminha cópias integrais de processos de apuração.
Senhor Superintendente, 1. De acordo com o § 4° do Art. 602 da Instrução Normativa n° 77/PRES/INSS, de 21/01/2015, encaminhamos, anexo, 01 (um) CD contendo as cópias integrais de 5 (cinco) processos de apuração de indícios de irregularidades, solicitando diligências desse Departamento de Polícia Federal no sentido de identificação do(s) responsável(eis) pelo recebimento dos benefícios, a seguir relacionados, após óbito dos respectivos titulares. N° BENEFICIO NOME DO TITULAR [redacted]
Atenciosamente, [redacted] c-a---ANDE
Gerente Executiva do INSS em Nat I-RN Matrícula [redacted]
PROTOCOLO 13 JUL. 2017 Thali ,ndes O?. NATAIJRN PK OLOISR/PF
RECEBID cnnnn Iflfl (O.fl ,•14C C,40 010. 084. 006. 025 [redacted]
[redacted] DESPACHO Para Inçuta› lopertgldellgnob
DPF# 'OPE é '1 Natal, itu os 1(9- DPF Osvaldo Scalezi Junior
DRCOR/SR/DPF/RN

Figura 1.2: Exemplo de um documento de NC. Um resumo extrativo do documento é feito extraindo e concatenando os textos de tarja amarela. Textos com listras pretas são informações privadas e por isso foram omitidos.

1.2 Relevância

Pesquisas anteriores sobre modelos de sumarização extrativa incluem avanços recentes feitos em modelos de PLN pré-treinados [35; 42; 35; 16], com enfâse na arquitetura BERT - um acrônimo para *Bidirectional Encoder Representations from Transformers*. Os modelos baseados nesta arquitetura alcançaram o estado-da-arte em diferentes tarefas de PLN, incluindo sumarização extrativa de documentos textuais. No entanto, boa parte dos modelos propostos recentemente para SA de texto sofrem de algumas limitações. Primeiramente, a maioria dos modelos não consegue lidar com documentos de tamanho arbitrário; esses modelos apenas processam os primeiros 512 *tokens*[1] de cada documento (entrada máxima permitida pelo modelo BERT) [17], o que pode produzir resultados insatisfatórios em documentos que possuem o comprimento maior do que este limite [17]. Além disso, modelos que conseguem lidar com comprimentos arbitrários requerem um retreinamento completo do modelo BERT pré-treinado. Treinar um modelo de rede neural tão complexo é custoso, em termos de processamento e memória e nem sempre é viável.

Outra limitação dos modelos de SA de texto a ser destacada é que a maioria deles não são capazes de funcionar bem em subdomínios de documentos de texto; geralmente, se especializam em um domínio específico. Por exemplo, um modelo que funciona para o domínio dos esportes tende a falhar no domínio político. No Capítulo 5, será mostrado que, em um conjunto de dados de NC, essas limitações supracitadas podem impactar negativamente no treinamento dos modelos e, consequentemente, na eficácia dos resumos gerados.

Sendo assim, faz-se necessária uma abordagem de sumarização extrativa de documentos textuais que i) considere como entrada textos de tamanho arbitrário, ii) consiga lidar de forma eficaz com documentos de diferentes domínios, iii) apresente boa precisão, ou seja, minimize erros de extração de partes não relevantes do texto.

1.3 Objetivos

Para promover a elaboração desta pesquisa, a seguinte hipótese geral foi considerada: *a utilização de abordagens de AM que lidam com documentos de tamanho arbitrário e perten-*

[1] *tokens* representam palavras ou sub-palavras

centes a diferentes subdomínios melhora a eficácia da predição das partes mais importantes dos documentos em comparação a modelos que não lidam com essas questões.

Com base nisso, e visando solucionar os desafios relacionados aos documentos das NCs da Polícia Federal, o objetivo geral deste trabalho é propor abordagens para sumarização extrativa com base no modelo BERTSUM [35], modelo de última geração em sumarização extrativa, que se baseia em BERT. As abordagens diferem das pesquisas existentes ao propor uma solução que possa lidar com documentos de tamanho arbitrário e domínios variados. Essas abordagens superam a limitação do BERTSUM ao incorporar a capacidade de capturar informações sequenciais ilimitadas e, assim, permitir o processamento de um texto arbitrariamente longo. Além disso, também é proposta uma abordagem para lidar com sumarização de texto multidomínio. Essa abordagem aproveita a técnica de agrupamento de textos baseada em similaridade textual para dividir os documentos em subdomínios e treinar um modelo em cada agrupamento formado. A ideia é especializar os modelos em cada subdomínio, obtendo resultados mais precisos. Todas as abordagens propostas são avaliadas em dois conjuntos de dados, um conjunto de dados de domínio privado (conjunto de dados de NC) e outro de domínio público (conjunto de dados de Wikihow). Ambos os conjuntos de dados são multidomínio.

1.4 Contribuições

Neste trabalho, são apresentadas três abordagens para sumarização extrativa de documentos textuais. As abordagens propostas utilizam como base o modelo BERTSUM e modificam esse modelo para poder lidar com documentos de tamanho arbitrário e multi-domínios. Essas abordagens serão descritas na Seção 4. As abordagens propostas foram avaliadas teórica e experimentalmente (usando dois conjuntos de dados do mundo real), quanto à eficácia (qualidade dos resumos gerados) e eficiência (tempo de execução). Assim, as principais contribuições deste trabalho são:

- Geração e avaliação dos modelos em um conjunto de dados brasileiro diversificado em grande escala com vários estilos de escrita e diferentes domínios, conveniente para treinar modelos complexos de sumarização de texto;

- Criação de uma abordagem baseada no modelo BERT para sumarização de texto em documentos em português do Brasil;
- Criação de uma abordagem capaz de lidar com documentos de tamanho arbitrário (BERTSUM-ALD);
- Criação de uma abordagem baseada no modelo BERT para lidar com documentos de tamanho arbitrário e com sumarização de texto de diferentes domínios usando a técnica de agrupamento (BERTSUM-ALD-MD);
- Criação de uma abordagem baseada no modelo BERT e com a utilização da técnica de *ensemble* para lidar com sumarização de textos (BERTSUM-ALD-ES);
- Avaliação das abordagens propostas usando métricas ROUGE em dois conjuntos de dados diferentes.

Esta pesquisa faz parte do Projeto de P&D ePol, desenvolvido pelo laboratório SPLab da UFCG em parceria com a Polícia Federal do Brasil. Até o presente momento, os seguintes indicadores de pesquisa foram obtidos:

- Apresentação desta pesquisa no Workshop de Teses e Dissertações do Simpósio Brasileiro de Banco de Dados (WTDBD 2021);
- Submissão de artigo científico intitulado "*Leveraging BERT for Extractive Text Summarization on Federal Police Documents*" ao *Journal of Knowledge and Information Systems* (2022).

1.5 Organização do trabalho

A estrutura deste documento está organizada da seguinte forma. No Capítulo 2, é apresentada a fundamentação teórica necessária para compreender o conteúdo do trabalho, como os conceitos relacionados à sumarização extrativa e os conceitos de aprendizagem de máquina. No Capítulo 3, são apresentados os trabalhos relacionados. No Capítulo 4, é apresentada a formalização do problema e as abordagens propostas para sumarização extrativa de documentos textuais, incluindo o uso da técnica de aprendizagem incremental. No Capítulo 5, é

apresentada a avaliação experimental das abordagens propostas, seguida de uma discussão sobre os resultados. Por fim, no Capítulo 6, são apresentadas as conclusões da pesquisa e as perspectivas para trabalhos futuros.

Capítulo 2

Fundamentação Teórica

Para fornecer a base necessária e para uma melhor compreensão das abordagens para o resumo automático de documentos textuais propostas nesta pesquisa, este capítulo apresenta uma visão geral das técnicas, arquiteturas e conceitos comumente usados. Na Seção 2.1, são apresentados os conceitos fundamentais sobre modelos de aprendizagem de máquina (AM) e algumas técnicas associadas. Na Seção 2.2, são apresentados os conceitos sobre PLN. Na Seção 2.3, são apresentadas as considerações finais do capítulo.

2.1 Conceitos de Aprendizagem de Máquina

Esta seção apresenta conceitos relacionados a AM, como a definição dos modelos, modelos pré-treinados e técnicas para combinar modelos (*ensemble*).

2.1.1 Definição de Aprendizagem de Máquina

A AM é uma subárea da Inteligência Artificial (IA). Os modelos de AM são definidos como modelos que analisam dados, identificam padrões e então usam esses padrões para realizar melhor sua tarefa atribuída [25]. Os quatros principais modelos de AM são aprendizagem supervisionada, aprendizagem não supervisionada, aprendizagem semi-supervisionada e aprendizagem por reforço. Nesta pesquisa, são utilizadas técnicas de aprendizagem supervisionada e não supervisionada, pois se adequam melhor a tarefa de SA de texto, as quais serão descritas mais detalhadamente.

Aprendizagem Supervisionada

Na aprendizagem supervisionada, os modelos recebem dados rotulados (cada instância de dados possui o seu valor real y_i), capturam a relação entre variáveis descritivas (dados) e uma variável alvo. Esses modelos usam como base fórmulas matemáticas para aprender uma função ideal $f : X \rightarrow Y$, que melhor representa o problema. Por exemplo, no contexto de SA de documentos textuais, o conjunto X representa as variáveis de entrada (por exemplo, sentenças de um documento) e o conjunto Y representa o valor a ser predito (por exemplo, "a sentença deve fazer parte do resumo"ou "a sentença não deve fazer parte do resumo").

O objetivo dos modelos de aprendizagem supervisionada é aprender uma função $g : X \rightarrow Y | g \approx f$, através dos dados de treinamento extraídos de um conjunto de dados. Esse conjunto de dados geralmente é dividido em três: treinamento, validação e teste. Os conjuntos de treinamento e validação são utilizados para para gerar o modelo preditivo e os dados de teste são utilizados para validar este modelo gerado (o conjunto de teste simula dados do mundo real aos quais o modelo não foi exposto no treinamento). A função g é comumente referenciada como sendo o modelo. Existem algumas variações de modelos (por exemplo, regressão linear e *multilayer perceptron*) que podem ser utilizados em diferentes cenários, dependendo do problema.

Aprendizagem Não Supervisionada

Na aprendizagem não supervisionada, os modelos recebem dados não rotulados, extraem padrões de semelhança anteriormente desconhecidos e tomam decisões com base na presença (ou ausência) de tais semelhanças em cada novo dado [26]. O agrupamento de dados é um exemplo clássico de um problema de aprendizagem não supervisionada, em que o modelo encontra pontos de dados semelhantes dentro de um conjunto de dados e agrupa-os de forma adequada (criando *clusters*) [6].Outros exemplos de aplicações de aprendizado não supervisionados são sistemas de recomendação de filmes ou músicas, detecção de anomalias e visualização de dados.

Agrupamento

O agrupamento de dados é uma técnica que visa fazer agrupamentos automáticos de dados, levando em consideração o grau de semelhança. Essa técnica tem como objetivo agrupar através de aprendizado não supervisionado exemplos de um conjunto de dados em n grupos, também denominados *clusters*. Uma técnica de agrupamento comumente utilizada é o *K-Means* [1]. Esta técnica tem como ideia encontrar itens semelhantes um com os outros, e mais distintos possíveis dentre os membros de outros grupos de acordo com seus atributos [1]. O funcionamento da técnica é baseada em cálculos de distância e média dos pontos (exemplos) até os centroides para poder definir uma posição clara entre os grupos, ou seja, a qual grupo cada exemplo pertence.

Na Figura 2.1 é ilustrado o passo à passo do algoritmo de agrupamento *K-Means*. O processo executado pelo *K-Means* é composto por quatro etapas. A primeira etapa é a inicilização, onde o algoritmo gera de forma aleatória k centroids, onde o número de centroids é representado ao parâmetro k. Estes centroids são pontos de dados que serão utilizados de pontos centrais dos grupos. A segunda etapa é atribuição ao grupo (*cluster*), onde é calculado a distância entre todos os pontos de dados e cada um dos centroids. Cada dado será atribuído ao centroid que tem a menor distância. Uma vez que os pontos de dados foram atribuídos aos centrois ou grupos conforme sua distância, a próxima etapa é a movimentação de centroids. Nesta etapa é calculada a média dos valores dos pontos de dados de cada grupo e o valor médio será a nova posição centróide. Na última etapa as etapas de atribuição ao grupo e movimentação de centroids são repetidas até os grupos se tornarem estáticos ou algum critério de parada tenha sido atingido.

Aprendizagem Incremental

Na aprendizagem incremental, ao contrário da aprendizagem de máquina clássica onde os modelos são treinados em lote e não integram continuamente novas informações ao modelo já construído, são aplicadas técnicas para incluir continuamente novos dados ao modelo treinado e, dessa forma, atualizar o modelo de acordo com os novos dados recebidos [4].

Ao lidar com problemas dinâmicos do mundo real, é necessário aplicar técnicas de aprendizagem incremental para que os modelos gerados não se tornem obsoletos, visto que as

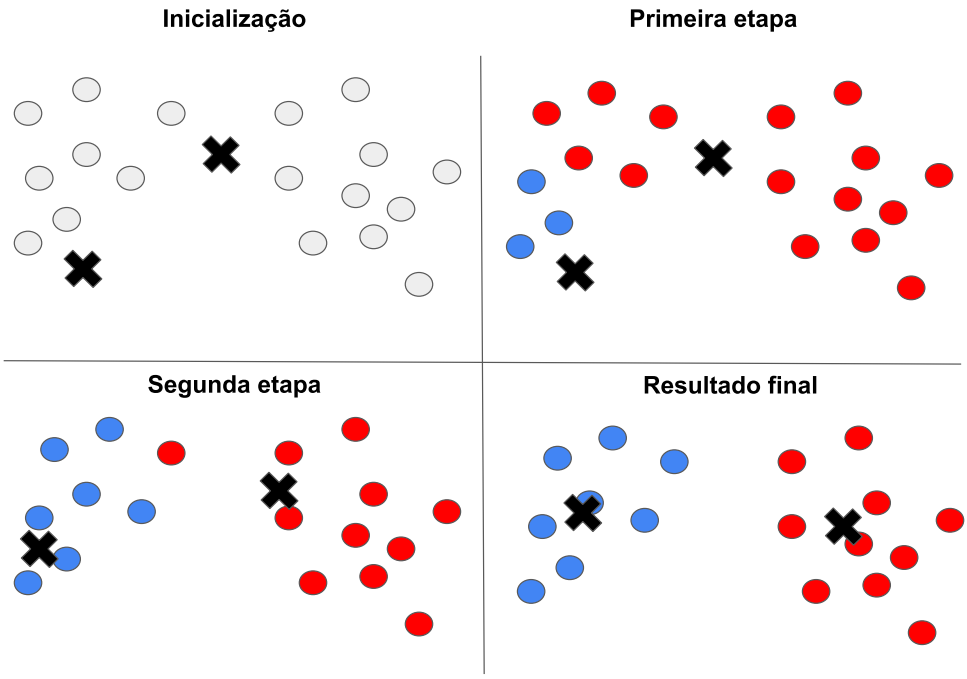


Figura 2.1: Funcionamento do algortimo de agrupamento *K-Means*.

técnicas clássicas de aprendizagem de máquina reconstroem novos modelos a partir do zero. Isso não apenas pode consumir mais tempo e recursos para treinamento dos modelos, mas também pode levar a modelos potencialmente desatualizados [36].

2.1.2 Abordagem *Ensemble*

Um *ensemble* combina vários modelos-base (base learners) de AM em um único modelo para obter um melhor desempenho preditivo, menor variância e viés, do que poderia ser obtido a partir de qualquer um dos modelos de aprendizagem constituintes sozinho. Pesquisadores mostraram que, em geral, a combinação de classificadores simples apresenta resultados mais acurados do que qualquer um dos classificadores usados individualmente [53]. A justificativa é que a diversidade dos classificadores base, seja em relação à arquitetura (parâmetros dos modelos, por exemplo), ao conjunto de características utilizadas ou aos dados de treinamento, permite que cada classificador aprenda bem uma parte diferente do espaço de dados e, assim, a combinação de todos eles gere um único classificador acurado. Os modelos-base podem ter estruturas diferentes, serem treinados com diferentes subamostras de dados e com-

binados de maneiras diferentes. O resultado do *ensemble* é a combinação das predições dos modelos que o compõe, sendo essa combinação realizada seguindo diferentes abordagens. O modelo de *ensemble* é uma solução para superar os seguintes desafios técnicos de construção de um único modelo:

- **Alta variância**: o modelo é muito sensível às entradas fornecidas para os padrões aprendidos, ou seja, o modelo tem alta variabilidade das predições;
- **Baixa acurácia**: um modelo ao tentar se ajustar a todos os dados de treinamento pode não ser bom o suficiente para atender às expectativas;
- **Apresenta ruído**: o modelo está fortemente dependente de uma ou algumas *features* ao fazer uma predição;
- **Alto viés**: o viés está relacionado à habilidade do modelo em se ajustar ao conjunto de dados, ou seja, se o modelo não se ajustou bem ao conjunto de dados, o modelo tem um alto viés.

Em geral, os modelos-base comumente utilizados são classificados como *weak learners*, ou seja, modelos com esquemas de aprendizado simples [67]. O oposto são *strong learners*, ou seja, modelos mais robustos, criados para alcançar alta eficácia nos dados de teste. Exemplos de técnicas de ensemble são: *Bootstrap aggregation (Bagging)*, *Boosting* e *Stacking*. Nesta pesquisa, é aplicada uma técnica de *Bagging* em uma das abordagens propostas. Essa técnica foi escolhida pela capacidade de paralelização do treinamento dos modelos-base e pela característica de reduzir a alta variância dos modelos. Essa técnica de *ensemble* é descrita a seguir.

Abordagem *Ensemble Bagging*

A abordagem *Ensemble Bagging* refere-se à combinação de n modelos-base (geralmente do mesmo tipo) treinados em paralelo e utilizando diferentes subconjuntos da base de dados, escolhidos aleatoriamente com reposição (amostragem *bootstrap*). Esta técnica é exemplificada na Figura 2.2. A base de dados é dividida em amostras com repetição e cada modelo-base é treinado em uma amostra criada. Por fim, as predições dos modelos são combinadas para gerar a predição final.

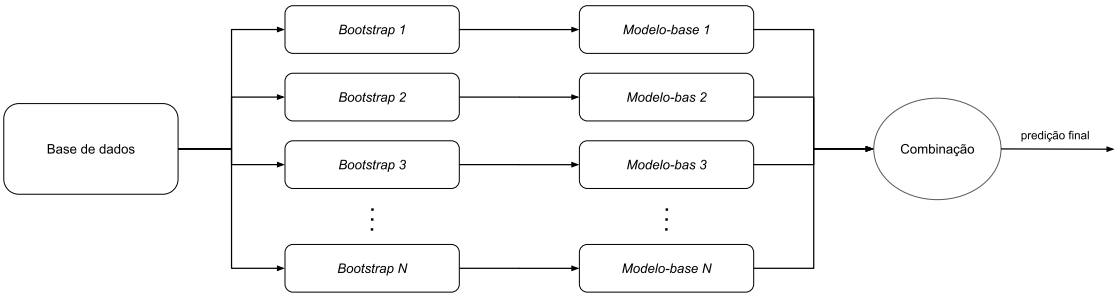


Figura 2.2: Exemplo de *ensemble* com abordagem *Bagging*

2.2 Conceitos de Processamento de Linguagem Natural

Esta seção apresenta conceitos relacionados ao PLN e SA. PLN é uma subárea da AM com objetivo de dar aos computadores a capacidade de compreender texto e palavras faladas de forma similar aos seres humanos [38]. Em particular, como programar modelos computacionais para processar e analisar grandes quantidades de dados de linguagem natural.

A PLN combina linguística computacional - modelagem baseada em regras da linguagem humana - com modelos aprendizado de máquina. Quando combinadas, essas técnicas permitem que os modelos computacionais processem a linguagem humana na forma de texto e "compreendam"seu significado, incluindo a intenção e o sentimento do escritor [38].

2.2.1 Análise Sintática e Semântica de Documentos Textuais

A análise sintática (sintaxe) e a análise semântica são as duas principais técnicas que levam à compreensão da linguagem natural. A sintaxe é a estrutura gramatical do texto, visa a determinar quais combinações de palavras são bem formadas em determinada língua. A análise sintática é o processo de análise da linguagem natural com as regras de uma gramática formal. As regras gramaticais são aplicadas a categorias e grupos de palavras, não a palavras individuais. Por outro lado, a semântica é o significado que está sendo transmitido. A semântica investiga as propriedades do significado bem como o estudo do significado das expressões das línguas naturais. A análise semântica é o processo de compreensão do significado das palavras e estrutura da frase. Isso permite que os computadores entendam

parcialmente a linguagem natural da mesma forma que os humanos.

2.2.2 Pré-processamento de Documentos Textuais

Esta subseção apresenta conceitos comumente utilizados no pré-processamento de documentos textuais na área de PLN. Antes de utilizar os dados textuais para geração de modelos de aprendizado de máquina, faz-se necessário processá-los. Geralmente este é o primeiro passo dos projetos de PLN. Algumas das etapas de pré-processamento são descritas nas subseções 2.2.2, 2.2.2, 2.2.2:

Tokenization

Nesta etapa, o texto é dividido em unidades menores. Pode ser utilizada tokenização de frase ou tokenização de palavra com base na declaração do problema. Por exemplo, o texto $d = \ 'texto\ a\ ser\ tokenizado.'$ pode ser tokenizado em palavras produzindo o resultado $r = ['texto', 'a', 'ser', 'tokenizado', '.']$.

Stemming

É conhecida como a etapa de padronização de textos em que as palavras são reduzidas para seu radical. Por exemplo, utilizando o Stemming palavras como $'aprender'$, $'aprendeu'$, $'aprendizado'$ serão convetidas para $'aprend'$.

Lemmantization

Essa é etapa é similar à etapa de Stemming, mas garante que ela não perca seu significado. A *Lemmantization* possui um dicionário pré-definido que armazena o contexto das palavras e verifica a palavra no dicionário para reduzi-la a sua forma raiz. Por exemplo, utilizando a *Lemmantization* palavras como $'encontrei'$, $'encontraram'$, $'encontraro'$, $'encontrariam'$ serão convertidas para palavra $'encontrar'$.

2.2.3 Sumarização Automática de Documentos Textuais

A SA de documentos textuais é uma subárea do PLN que visa à produção automática de sumários (resumos) a partir de um ou mais textos fontes. Os sumários podem ser produzidos

por diversas técnicas. Nesta seção, apresentam-se as duas técnicas de sumarização (extrativa e abstrativa), como ilustrado na Figura 2.3, onde a técnica extrativa gera o sumário através da seleção de sentenças do documento e técnica abstrativa gera um novo texto contendo as informações mais relevantes do documento.

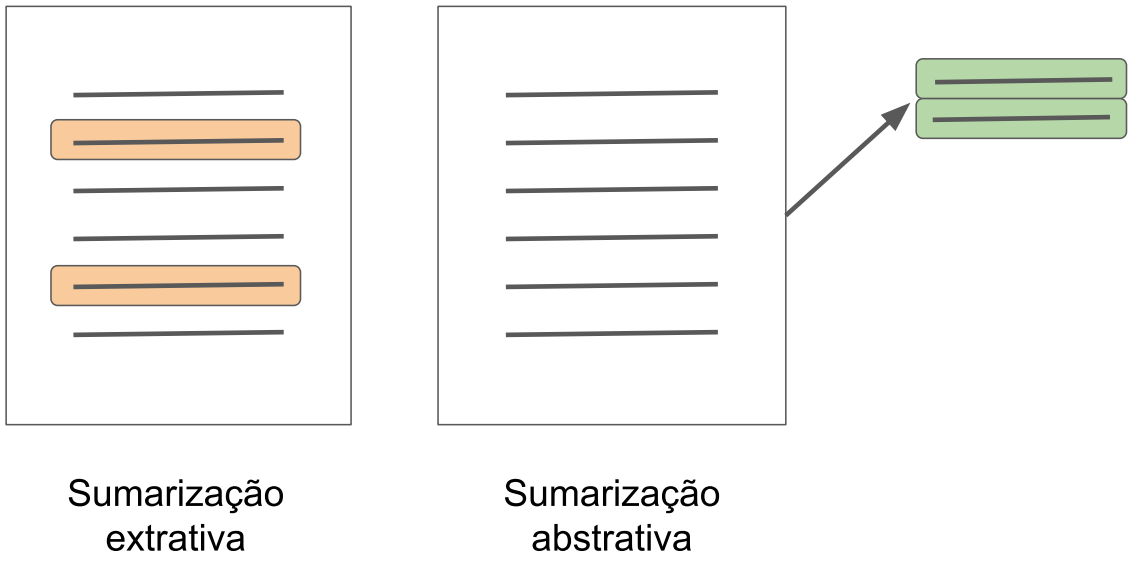


Figura 2.3: Comparação entre a sumarização extrativa e abstrativa

2.2.4 Sumarização Extrativa

A tarefa de sumarização extrativa de texto, cujo processo é apresentado resumidamente na Figura 2.4 para o resumo automático de documentos, pode ser considerada como um problema de classificação em AM [21]. A primeira etapa consiste em pré-processar o documento textual. As tarefas de pré-processamento incluem: segmentação do texto em sentenças e palavras, remoção de *stop words* (palavras que geralmente ocorrem no texto mas que tem pouco

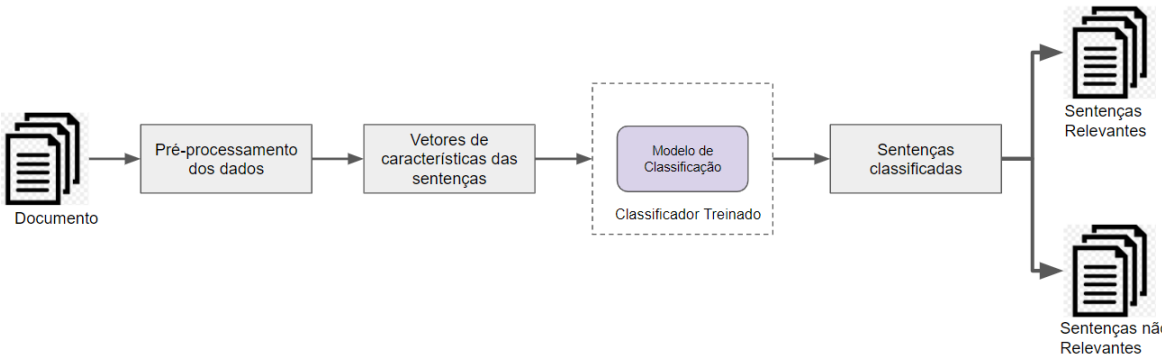


Figura 2.4: Fluxo do processo de sumarização extrativa

sentido semântico, (e.g. como, para, as, e, os, de, com, sem), entre outras. Em seguida, são extraídas características das sentenças, incluindo tanto características sintáticas como, por exemplo, o tamanho da sentença e sua posição no documento (início, meio ou fim), quanto características semânticas como, por exemplo, similaridade entre uma sentença e as demais. Tais características servem como entrada para o modelo de classificação (Classificador) decidir se a sentença é uma sentença importante ou não. Por fim, as sentenças importantes são extraídas e concatenadas a fim de criar o resumo final.

2.2.5 Sumarização Abstrativa

A tarefa de sumarização abstrativa de texto, cujo processo é apresentado resumidamente na Figura 2.5, consiste em reescrever o texto original de forma reduzida, mantendo os pontos mais relevantes. O processo é semelhante ao modo como os humanos leem um texto e o resumem em suas próprias palavras. Isso requer modelos complexos que dependem de métodos linguísticos para ter uma compreensão profunda do conteúdo do texto [62; 46; 71]. Como o método de sumarização abstrativa gera um resumo baseado na compreensão semântica dos documentos originais, nem todas as palavras do resumo aparecem nos documentos originais.

2.2.6 Modelos pré-treinados

Um modelo pré-treinado, geralmente, foi previamente treinado em um grande conjunto de dados (e.g. dados da Wikipédia). O modelo pré-treinado pode ser ajustado para uma tarefa

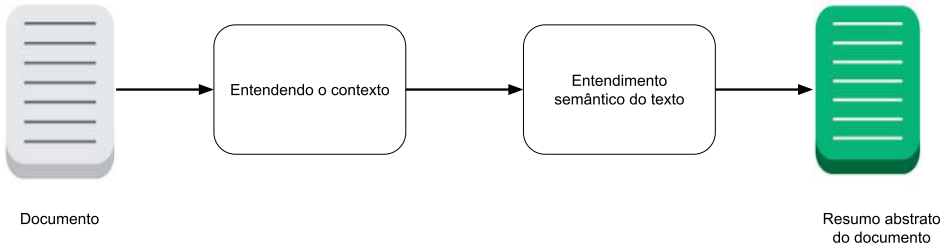


Figura 2.5: Fluxo do processo de sumarização abstrativa

específica. Em vez de construir um modelo do zero para resolver um problema semelhante, o modelo pode ser utilizado na aprendizagem do outro problema como ponto de partida [74]. Por exemplo, a maioria dos modelos pré-treinados em PLN aproveitam a ordem natural do texto nos documentos [59]. O word2vec [41] usa as palavras ao redor dentro de uma janela de tamanho fixo para prever a palavra no meio.

Transformer

É a tendência da arquitetura pré-treinada para muitas tarefas de PLN. Até 2017, a Rede Neural Recorrente (RNR) era a abordagem principal para muitas tarefas de PLN. O problema com o uso de RNRs é que, por não ser uma arquitetura pré-treinada (ou seja, o modelo precisa ser treinado do zero para cada nova tarefa), e também por recorrência em suas camadas, requerem grandes quantidades de dados e elevados recursos de computação para se ajustar aos dados. Além disso, tendem a sofrer de baixa eficácia com sequências muito longas e estão sujeitas à sobreajuste (*overfitting*) [65]. Em 2017, o trabalho proposto por [65] apresentou uma arquitetura superior denominada Transformer, que elimina a necessidade do uso de RNRs em favor do uso de uma arquitetura composta por redes *feed-forward* e o mecanismo de atenção. A ideia por trás do mecanismo de atenção é permitir que o modelo utilize as partes mais relevantes da sequência de entrada de maneira flexível, por uma combinação ponderada de todos os vetores de entrada, sendo atribuídos os maiores pesos aos vetores mais relevantes, ou seja, dando mais atenção as partes mais relevantes do texto.

Arquitetura *Transformer.*

A arquitetura *Transformer* é aprensentada na Figura 2.6. Essa arquitetura é composta por um componente de codificação (lado esquerdo da Figura 2.6), um componente de decodificação (lado direito da Figura 2.6) e as conexões entre eles. O componente de codificação é uma pilha de codificadores (o artigo original empilha seis condificadores [20]). O componente de decodificação é uma pilha de decodificadores de mesma quantidade. Os codificadores são todos idênticos em estrutura, mas não compartilham os pesos das redes neurais. Cada codificador é dividido em duas subcamadas: uma camada de autoatenção e uma camada de rede neural. Na saída de cada camada, existem duas etapas (Adição e Normalização). Na adição, é somada a saída de uma camada com a entrada ($F(x) + x$). A ideia foi introduzida por [22] com o modelo ResNet. Esta ideia é uma das soluções para o problema da dissipação do gradiente que dificulta o treinamento de redes neurais profundas. A etapa de normalização da camada [3] é outra forma de normalização de redes neurais. Esta etapa é uma das etapas computacionais para facilitar o treinamento do modelo, melhorando assim o desempenho e o tempo de treinamento.

O decodificador tem ambas as camadas do codificador, mas entre elas há uma camada de atenção que ajuda o decodificador a se concentrar em partes relevantes da sentença de entrada. Por fim, cada codificador se conecta com todos os outros decodificadores da arquitetura.

Com o objetivo de demonstrar sua eficácia, essa arquitetura foi aplicada a uma tarefa de tradução, obtendo, na época, o estado-da-arte na tradução de texto da língua inglesa para a francesa.

Modelo BERT

No final de 2018, pesquisadores do Google introduziram uma arquitetura de aprendizagem não supervisionada baseada na arquitetura *Transformer* chamada BERT. BERT é um modelo de pré-treinamento de PLN, foi projetado para pré-treinar representações bidirecionais profundas de texto não rotulado, condicionando conjuntamente o contexto da esquerda e da direita. O BERT obteve resultados estado-da-arte em muitas tarefas de PLN como, por exemplo, análise de sentimentos [17].

Para a maioria das tarefas de PLN, o BERT concatena diferentes partes da entrada em

uma sequência que começa com o token [CLS] e insere o token [SEP] entre duas partes diferentes. Antes de passar os dados pelas camadas do *Transformer*, o BERT une três vetores (*embeddings*) diferentes em um (incorporação de palavras, incorporação de posições e incorporação de segmentos) [17].

Modelo BERTSUM

BERTSUM é uma extensão do BERT na tarefa de sumarização extrativa de texto. O modelo utiliza apenas os primeiros 512 tokens do documento textual como entrada [35]. Para classificar as sentenças, o modelo BERTSUM adiciona [CLS] ao início de cada sentença e [SEP] ao final de cada sentença, indicando o final dessa sentença, conforme ilustrado na Figura 2.8. A camada de sumarização pontua cada vetor de [CLS] que representa o sentido semântico da sentença. Finalmente, as sentenças com as três maiores pontuações compõem o resumo. No artigo [35], foram propostas três camadas de sumarização (classificador simples, *Transformer* inter-frases e RNR). Nos resultados experimentais apresentados no artigo [35], todas as camadas de sumarização apresentaram resultados semelhantes em termos de eficácia. Por questões de simplificação, é utilizado apenas o classificador simples como a camada de sumarização nas abordagens propostas nesta pesquisa.

Mesmo que o modelo BERTSUM tenha alcançado o estado-da-arte em sumarização extrativa de texto, foi evidenciado, nos experimentos desta pesquisa, que a limitação da utilização de apenas os 512 primeiros tokens do documento, aplicada pelo BERTSUM, resulta na perda de informações importantes em documentos extensos. Sendo assim, se as sentenças-chave (sentenças mais importantes do documento) estiverem localizadas no final do documento, essas sentenças nunca farão parte do resumo gerado. Para resolver esta limitação, nesta pesquisam foram propostas diferentes abordagens que conseguem lidar com documentos de comprimento arbitrário.

2.2.7 Métricas para Avaliação da Qualidade de Resumos

Para avaliar os resultados dos algoritmos de SA de texto, são utilizadas métricas de qualidade. A qualidade refere-se à semelhança entre o resumo gerado pelos algoritmos e o resumo de referência. No contexto de sumarização, a ROUGE (*Recall-Oriented Understudy*

for Gisting Evaluation) [32] é uma métrica que vem ganhando bastante popularidade entre as pesquisas sobre sumarização. ROUGE mede a qualidade sintática de resumos calculando unidades lexicais sobrepostas, como unigrama (ROUGE-1), bigrama (ROUGE-2), trigrama (ROUGE-3) e a subsequência comum mais longa (ROUGE-L). Para explicação das métricas considere:

Métrica ROUGE

ROUGE é um conjunto de métricas utilizado para avaliação automática de sumarização em PLN [33]. As quatro métricas de avaliação mais utilizadas são:

ROUGE-1: refere-se à sobreposição de 1 grama (cada palavra) entre os resumos produzidos pelo modelo avaliado e pelo modelo de referência;

ROUGE-2: refere-se à sobreposição de bigramas entre os resumos produzidos pelo modelo e o de referência;

ROUGE-N: refere-se à sobreposição de N gramas (cada uma sendo uma palavra) entre os resumos produzidos pelo modelo e o de referência;

ROUGE-L: refere-se à estatística baseada em Subsequência Comum Mais Longa (*Longest Common Subsequence* - LCS) [3]. O problema de LCS leva em consideração a similaridade da estrutura do nível da sentença e identifica a co-ocorrência mais longa em n-gramas em sequência.

Nesta pesquisa, os modelos foram avaliados utilizando as pontuações de ROUGE-1, ROUGE-2 e ROUGE-L com os resumos de referncia para avaliar a eficácia de diferentes métodos e nos concentrarmos na pontuação $F_{measure}$ (F1), conforme mostrado na Equação 2.3 (a métrica F1 é baseada nas Equações 2.1 e 2.2).

$$P_{ROUGE-N} = \frac{\sum_{gram_n \in ReferenceSummary} Count_{match}(gram_n)}{\sum_{gram_n \in CandidateSummary} Count(gram_n)} \tag{2.1}$$

$$R_{ROUGE-N} = \frac{\sum_{gram_n \in ReferenceSummary} Count_{match}(gram_n)}{\sum_{gram_n \in ReferenceSummary} Count(gram_n)} \tag{2.2}$$

$$F_{ROUGE-N} = \frac{(1+B^2) R_{ROUGE-N} P_{ROUGE-N}}{R_{ROUGE-N} + B^2 P_{ROUGE-N}} \tag{2.3}$$

onde n representa o comprimento do n-grama (ou seja, ROUGE-1, ROUGE-2), $gram_n$ é o número de n-gramas no texto e $Count_{match}(gram_n)$ é o número máximo de n-gramas

co-ocorrendo em um resumo gerado e um resumo de referência. Por exemplo, na Figura 2.9, é ilustrado um exemplo da utilização da métrica ROUGE, onde foram comparadas duas sentenças e calculada a pontuação ROUGE-1. As duas frases são um resumo de referência e um resumo gerado automaticamente. Primeiro, as sentenças foram divididas em *tokens* de n-gram (neste caso, os textos forma divididos em *tokens* de unigrama, pois a métrica que está sendo avaliada é o ROUGE-1). Em seguida, são verificadas quais n-gramas co-ocorrem em ambos os textos. Depois disso, são aplicadas as Equações 2.1 e 2.2 para calcular a precisão e a cobertura, e então é medida a pontuação F1. Por fim, o β (B) representa um parâmetro de configuração. Um valor beta menor, como 0,5, dá mais peso à precisão e menos ao cobertura, enquanto um valor beta maior, como 2,0, dá menos peso à precisão e mais peso a cobertura no cálculo da pontuação.

$$P_{lcs} = \frac{LCS(X,Y)}{l_X} \tag{2.4}$$

$$R_{lcs} = \frac{LCS(X,Y)}{l_Y} \tag{2.5}$$

$$F_{lcs} = \frac{(1+B^2)R_{lcs}P_{lcs}}{R_{lcs}+B^2P_{lcs}} \tag{2.6}$$

A métrica ROUGE-L é mostrada nas Equações 2.4, 2.5 e 2.6, onde X é uma sentença do resumo de referência, Y é uma sentença do resumo gerado, l_X é o comprimento de X, l_Y é o comprimento de Y e $LCS(X,Y)$ é o comprimento da subsequência comum mais longa entre X e Y. A métrica ROUGE-L calcula a maior subsequência comum entre um determinado texto e um texto de referência.

2.3 Considerações Finais

Neste capítulo, foram apresentados os principais conceitos relacionados às técnicas de AM e PLN. Em relação à AM, foram apresentados a definição de modelo, aprendizado supervisionado e não supervisionada, técnicas de ensemble, modelos pré-treinado, e modelos estado-da-arte. Além disso, foram descritos conceitos SA e técnicas de avaliação de resumos gerados por modelos de sumarização. A seguir, serão apresentados os trabalhos relacionados à sumarização automática de texto, com destaque para suas vantagens e limitações.

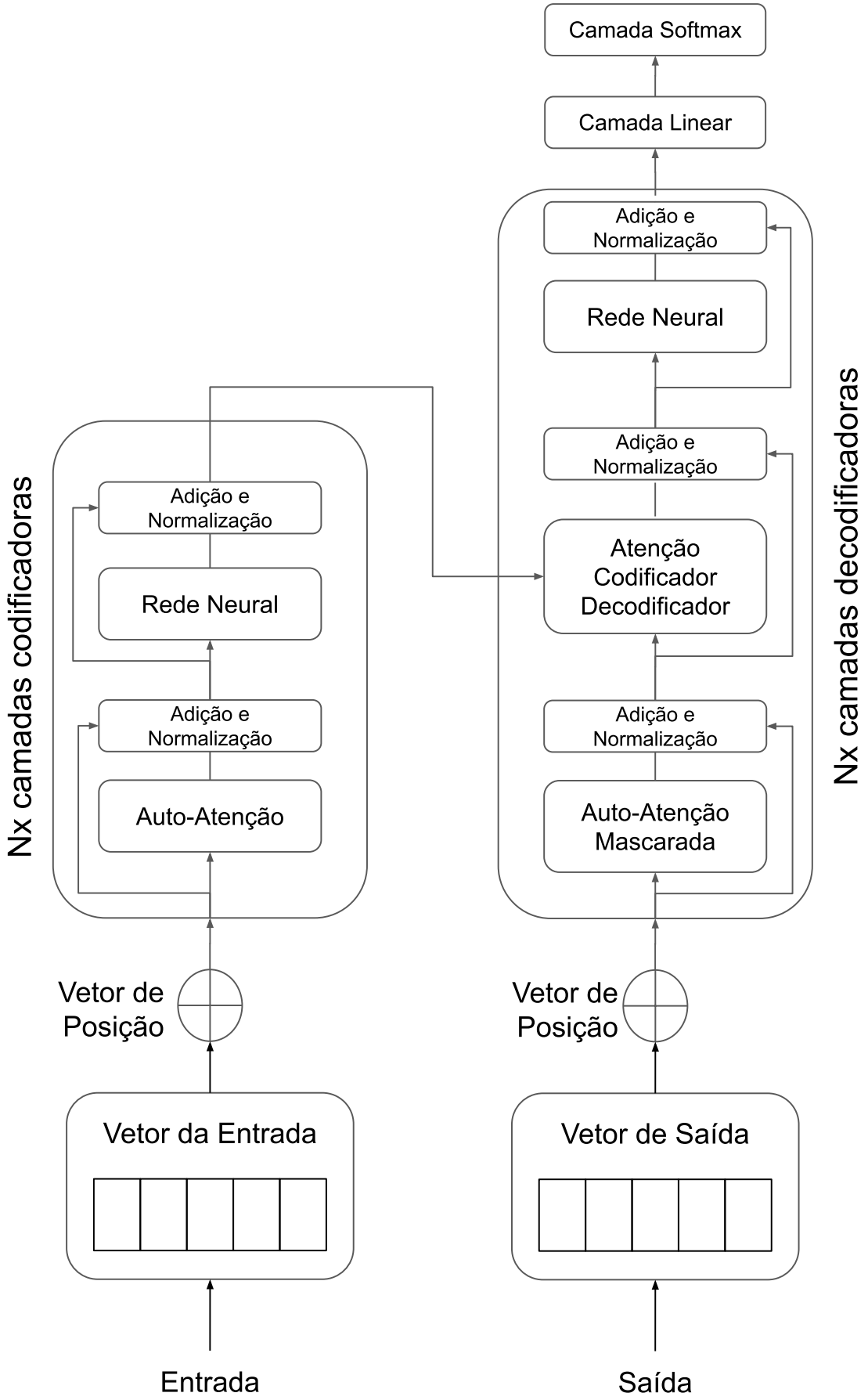


Figura 2.6: Arquitetura *Transformer*

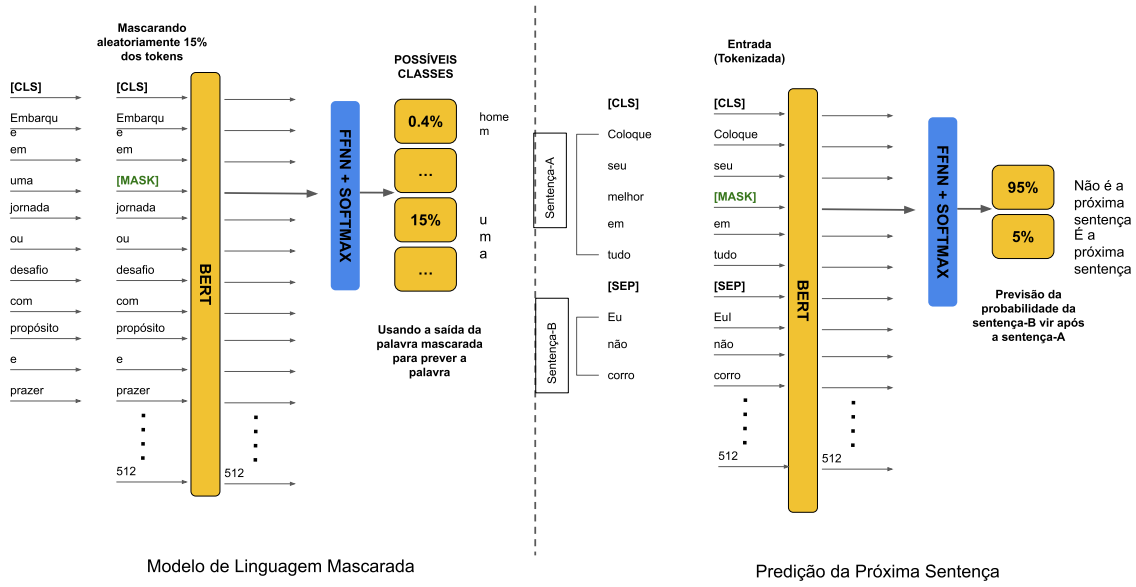


Figura 2.7: Predição de Linguagem Mascarada e Predição da Próxima Frase

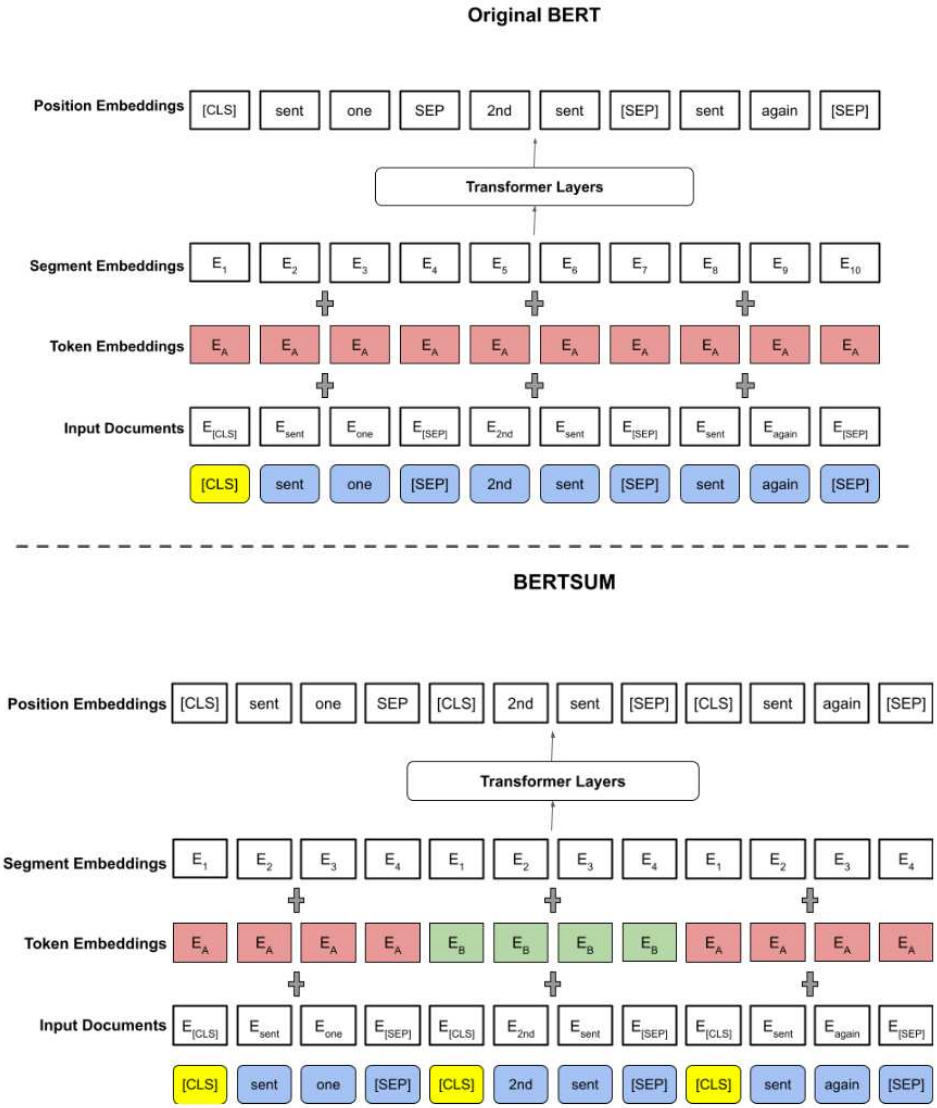


Figura 2.8: Arquitetura BERT: BERTSUM vs. BERT

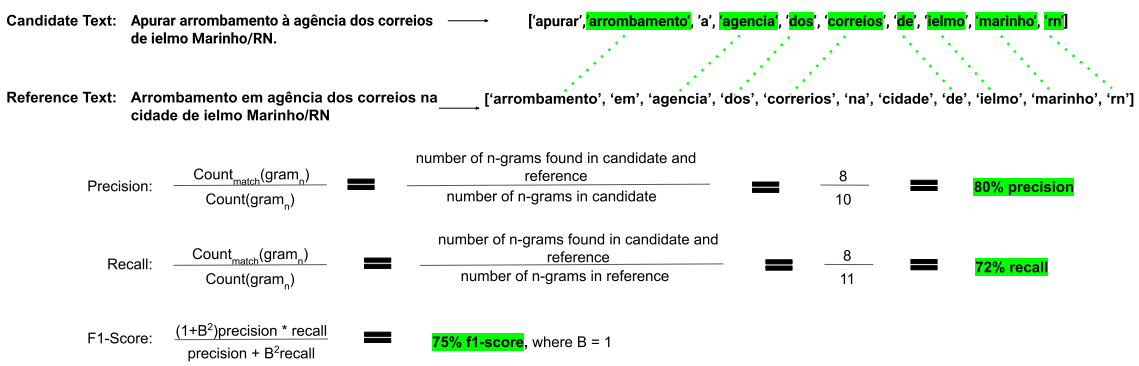


Figura 2.9: Exemplo da métrica ROUGE-1

Capítulo 3

Trabalhos Relacionados

Neste capítulo, é apresentada a metodologia empregada para busca por trabalhos que abordam o problema de SA de documentos textuais (Seção 3.1). Em seguida, na Seção 3.2, são discutidos e comparados os trabalhos relacionados à sumarização. Por fim, na Seção 3.3, são discutidas as considerações finais do capítulo.

3.1 Metodologia

Os trabalhos selecionados e estudados nesta pesquisa foram encontrados a partir da aplicação da abordagem *ad-hoc* (variando as chaves de busca) aos sites de pesquisa *IEEE Xplore Digital Library*[1], *ACM Digital Library*[2], *Google Scholar*[3] e *Papers with Code*[4]. As principais palavras-chave consideradas foram "SA","aprendizagem de máquina", "extrativa", "abstrativa", "português", "BERT", "Transformers", "policial", "notícia crime", em inglês.

Em seguida, foram selecionados para leitura os artigos que continham as palavras-chave, enquanto os demais foram descartados. Além disso, foi aplicada a técnica *snowball sampling* [48] que consiste em, a partir dos artigos selecionados e com base nas referências apresentadas nestes artigos, selecionar trabalhos adicionais para compor a pesquisa. São descritos a seguir os trabalhos selecionados e estudados nesta pesquisa.

[1] https://ieeexplore.ieee.org/Xplore/home.jsp

[2] https://dl.acm.org/

[3] https://scholar.google.com/

[4] https://paperswithcode.com/

3.2 SA de Texto

Desde meados do século passado, o problema de SA de texto tem sido estudado por pesquisadores, quando os autores de [37], usando um modelo estatístico simplificado, criaram um modelo capaz de gerar resumos de documentos textuais. Desde então, surgiram vários trabalhos relacionados a esse problema, alguns dos quais se concentram na determinação das características sintáticas das sentenças mais importantes do documento, enquanto outros focam nas características semânticas dos textos.

Vários esforços têm sido realizados no avanço da SA de texto usando diferentes modelos, tecnologias e ferramentas [27]. Antes de 2014, as pesquisas de sumarização focavam i) em métodos de sumarização que utilizam características definidas manualmente para classificar as sentenças e, assim, identificar as sentenças mais importantes em um documento [31; 60; 18; 51; 24] ou ii) na extração de sentenças de documentos usando modelos estatísticos e redes neurais simples, com pouco sucesso na extração das sentenças mais importantes dos documentos. Em [12], um modelo de sumarização extrativa é apresentado, com base em características sintáticas (e.g. posição da sentença no documento e tamanho da sentença), usando uma base de dados contendo documentos sobre esportes. Os experimentos foram conduzidos nos dados resultantes de vários jornais incluindo o New York Times e o *BBC*. A eficácia, baseada nas métricas de Precisão, Cobertura e $F_{measure}$ atingiu uma média de 75% em capturar as sentenças mais importantes dos documentos. A desvantagem desse modelo é que se baseia apenas em características sintáticas das sentenças. Na prática, o conteúdo semântico é mais importante para identificar se uma sentença deve fazer parte do resumo ou não.

Atualmente, com o grande volume de dados textuais disponíveis na Internet, muito mais esforços são necessários para realizar a SA de documentos textuais. A partir de 2014, com o desenvolvimento das redes neurais profundas, vários modelos baseados em redes neurais foram utilizados para tentar resolver os problemas de PLN [55]. O uso de redes neurais recorrentes (RNRs) dominou estudos e pesquisas na área de PLN, proporcionando melhor eficácia. O modelo de rede neural presente em [73] que promove um modelo pontuação e seleção conjunta de sentenças para sumarização extrativa de documentos obteve, na época, o estado-da-arte em sumarização extrativa. Além disso, com a introdução do mecanismo de

atenção, ou seja, um mecanismo que permite que redes neurais profundas tenham um entendimento mais preciso dos dados textuais [20], os modelos obtiveram grandes melhorias em seus resultados. Além disso, esse mecanismo possibilitou que as pesquisas em sumarização abstrativa alcançassem resultados de eficácia similares aos das abordagens de sumarização extrativa que, até esse momento, dominavam as pesquisas em sumarização. Porém, a utilização de RNRs, mesmo com o uso de LSTMs (*Long short-term memory*) e a implementação do mecanismo de atenção, ainda está sujeita a sofrer de baixa eficácia com sequências de dados muito longos [65]. Nesse contexto, em 2017, o trabalho [65] propôs uma arquitetura conhecida como *Transformers* que utiliza o mecanismo de atenção, porém sem a utilização de RNRs, para lidar com sequências de texto, ou seja, que não sofre das limitações inerentes as RNRs.

Modelos pré-treinados. A introdução de modelos de linguagem pré-treinados no área de PLN alavancou os limites da compreensão e geração da linguagem. A aplicação de modelos baseado na arquitetura *Transformers* em diferentes tarefas de PLN tornaram-se a principal tendência das pesquisas. O modelo (BERT) é um dos modelos baseado na arquitetura *Transformers*. É pré-treinado em uma enorme quantidade de dados (conjuntos de dados de pré-treinamento), sendo o BERT-Large treinado em mais de 2500 milhões de palavras. O BERT quando foi proposto alcançou o estado-da-arte em várias tarefas de PLN, como classificação de sentimentos, inferência de linguagem natural, semelhança textual semântica, entre outras. Após isso, surgiram diferentes pesquisas que propuseram variações do modelo BERT que mitigaram diferentes limitações do modelo.

O modelo BERT possui um mecanismo de auto-atenção completo. Isso leva a um crescimento quadrático dos requisitos computacionais e de memória para cada novo token de entrada. Sendo assim, os pesquisadores do BERT restringiram seu tamanho máximo de entrada para 512 tokens, o que significa que esse modelo não pode ser usado para entradas maiores e para tarefas como a sumarização de documentos grandes. Isso basicamente significa que um texto grande deve ser dividido em segmentos menores antes de aplicá-los como entrada. Essa fragmentação de conteúdo também causa uma perda significativa de contexto, o que torna sua aplicação limitada. Para mitigar essa limitação os pesquisadores em [69] propuseram o modelo BigBird. O BigBird usa um mecanismo de atenção esparsa, o que significa que o mecanismo de atenção é aplicado token por token, diferentemente do BERT,

onde o mecanismo de atenção é aplicado a toda a entrada apenas uma vez. Isso permitiu que o modelo superasse a dependência quadrática do BERT, preservando as propriedades do mecanismo de auto-atenção completo. Outro modelo que propôs uma alternativa para a limitação do mecanismo de auto-atenção completo do BERT foi o modelo LongForter proposto em [5]. O modelo Longformer introduz um mecanismo de atenção que cresce linearmente com o comprimento da sequência através da introdução de uma janela deslizante de tamanho w. Isso limita cada token a atender apenas a um subconjunto de todos os tokens - os locais considerados de maior importância. Embora esse padrão de atenção possa parecer limitado, ele ainda permite que uma rede de transformadores multicamadas tenha um campo receptivo que cubra toda a sequência. Ambos os modelos BigBird e Longformer promovem alternativas ao modelo BERT para lidar com longas sequências de texto.

Outra limitação do modelo BERT, que foi mitigado por outras pesquisas, está relacionado ao pré-treinamento de linguagem mascarada, que faz com que o BERT negligencia a dependência entre as posições mascaradas, assumindo que os tokens mascarados são independentes uns dos outros e obtendo uma discrepância no ajuste do pré-treinamento. Essa limitação foi estudada em [68] e os pesquisadores propuseram um novo modelo chamado XLNet. O modelo XLNet possui um treinamento de modelo de linguagem modificado que aprende distribuições condicionais para todas as permutações de tokens em uma sequência, ou seja, o XLNet aprende a prever cada palavra em uma sequência usando qualquer combinação de outras palavras nessa sequência. Dessa forma, está sendo apresentado contextos difíceis e às vezes ambíguos para inferir se uma palavra está ou não em uma frase. É isso que permite extrair mais informações do corpus de treinamento e melhorar os resultados do modelo.

Contudo, mesmo existindo atualmente modelos mais avançados que o modelo BERT. Esses modelos ainda não popularizaram como o modelo a ponto de existir uma variação treinada em um corpus na língua portuguesa. Dessa forma, esses modelos se limitam apenas a utilização em certas línguas. A utilização desses modelos exigiria o pré-treinamento do modelo, o que é computacionalmento custoso. Por fim, apenas o modelo BERT possui uma versão treinada em um corpus de texto em português. Sendo essa, a principal justificativa para a não utilização de modelos mais avançados neste trabalho.

Sumarização extrativa com BERT. A maior parte dos trabalhos de pesquisa mais re-

centes em sumarização extrativa de documentos textuais utilizam como base um modelo pré-treinado baseado na arquitetura *Transformers* chamado BERT para alavancar seus resultados. Seguindo o sucesso de arquiteturas baseadas em modelos pré-treinados para múltiplas tarefas, o modelo que obteve o estado-da-arte no conjunto de dados CNN/DailyMail [35] usa uma nova técnica que modifica a sequência de dados de entrada e os *embeddings* do BERT para que seja possivel extrair resumos. No trabalho [43], foi proposto um modelo não supervisionado onde os pesquisadores utilizaram como base o modelo BERT para geração dos *embeddings* das sentenças do texto e o modelo de agrupamento *K-Means* para identificação e extração das sentenças mais próximas dos centróides para geração do resumo.

Para realização de sumarização extrativa, a maioria dos modelos atuais concentra-se em resumos de documentos ao nível de sentença [49; 50]. Por outro lado, o artigo [72] cria uma mudança de paradigma no que diz respeito à forma como construímos modelos de sumarização extrativa. Em vez de gerar resumos extraindo as sentenças individualmente, a pesquisa formula a tarefa de sumarização extrativa como um problema de similaridade semântica entre textos, ou seja, são criados vários resumos com diferentes combinações de sentenças e a similaridade desses resumos com o documento original é avaliada.

Modelos de sumarização para lidar com documentos longos. Alguns trabalhos, por sua vez, se concentram em como lidar com a SA de documentos longos. Na pesquisa [11] foi proposto um sistema de sumarização abstrativo de texto para lidar com documento longos. O artigo [70] propõe uma arquitetura combinando *Transformer* e *Long short-term memory* (LSTM), que resolve o problema do modelo do *Transformer*, que não pode ser usado em textos muito longos. O modelo usa a arquitetura do *Transformer* para modelar apenas a dependência local e capturar recorrentemente a dependência global inserindo LSTMs em cada camada do BERT. O modelo proposto é nomeado como BERT-AL.

No trabalho [13], os autores propõem um novo modelo de sumarização abstrativa usando um codificador hierárquico que representa cada frase utilizando RNR ao nível de palavra e então processa todas as sentenças utilizando RNR ao nível de sentença. O modelo inclui um codificador hierárquico, que captura a estrutura semântica do documento e um decodificador que gera o resumo. O decodificador foca em diferentes partes do texto e permite que o modelo tenha uma representação mais acurada das informações importantes do texto, resultando em um melhor vetor de contexto semântico.

Em [34], os autores apresentam uma arquitetura somente com o decodificador da arquitetura *Transformer*, que pode lidar com sequências longas. O modelo primeiro seleciona grosseiramente um subconjunto da entrada utilizando sumarização extrativa e, em seguida, treina um modelo abstrativo baseado nesse subconjunto para geração do resumo. Esses trabalhos são semelhantes às abordagens presentes nesta pesquisa, que consistem em dividir um longo documento em vários subdocumentos para processá-los em paralelo e, em seguida, mesclá-los. Porém, esses trabalhos têm como limitação a necessidade de retreinar o modelo do zero, o que é custoso em termos de tempo e recursos computacionais. Além disso, nenhum desses trabalhos consegue lidar com documentos de texto de múltiplos domínios.

Resumo automático de textos em português. Em relação aos trabalhos relacionados ao resumo automático de textos em português, poucos trabalhos foram realizados nesta área [61; 15; 54; 64]. A maior parte das pesquisas existentes estão desatualizadas em relação aos métodos utilizados atualmente. Os trabalhos utilizaram modelos não supervisionados ou modelos estatísticos simples que, comparando aos métodos atuais, apresentam resultados inferiores. Esses modelos usam apenas a abordagem extrativa e se baseiam em características sintáticas das sentenças para identificar as partes mais importantes dos documentos. Por fim, não foram encontradas pesquisas de SA na área de documentos policiais, revelando-se uma área de pesquisa inexplorada.

Na Tabela 3.1, é apresentado um resumo dos trabalhos relacionados descritos anteriormente. Cada autor usa diferentes modelos de AM para sumarização automática de texto. A maioria deles usa a mesma base de dados para avaliação dos modelos (CNN/DailyMail). Apenas os autores [11; 50] utilizam uma técnica abstrativa de sumarização. Além disso, é visto ainda que os modelos extrativos em geral obtém resultados superiores aos modelos abstrativos.

3.3 Considerações Finais

Neste capítulo, foram apresentados os trabalhos relacionados à SA de documentos textuais. Em geral, os trabalhos utilizam modelos de AM e modelos pré-treinados para realizar a sumarização. Além disso, poucos trabalhos foram realizados em sumarização de textos em português e nenhum trabalho aplica a técnica de agrupamento automático para lidar com do-

cumentos de múltiplos domínios. Por fim, nenhum trabalho aplica a técnica de aprendizagem incremental para atualizar o modelo de sumarização.

No capítulo seguinte, serão apresentadas as soluções propostas para sumarização de documentos textuais, cujas principais diferenças são as abordagens utilizadas para lidar com documentos de tamanho arbitrário e de diferentes subdomínios.

Artigo	Modelo	Base de dados	Técnica	Eficácia
[73]	NeuSam	CNN/DailyMail	Extrativa	ROUGE-1=41.59 ROUGE-2=19.01 ROUGE-L=37.98
[35]	BERTSUM	CNN/DailyMail	Extrativa	ROUGE-1=43.25 ROUGE-2=20.24 ROUGE-L=39.63
[11]	DCA	CNN/DailyMail	Abstrativa	ROUGE-1=41.69 ROUGE-2=19.47 ROUGE-L=37.92
[43]	BERT-K-MEANS	Documentos de Palestras	Extrativa	Não informada
[49]	SummaRunner	CNN/DailyMail	Extrativa	ROUGE-1=39.60 ROUGE-2=16.20 ROUGE-L=35.3
[50]	T-CONVS2S	XSum	Abstrativa	ROUGE-1=31.89 ROUGE-2=11.54 ROUGE-L=25.75
[72]	MATCHSUM	CNN/DailyMail	Extrativa	ROUGE-1=44.41 ROUGE-2=20.86 ROUGE-L=40.55
[70]	BERT-AL	CNN/DailyMail	Extrativa	ROUGE-1=42.61 ROUGE-2=19.79 ROUGE-L=39.07

Tabela 3.1: Resumo dos trabalhos relacionados à sumarização automática de texto

Capítulo 4

Abordagens BERT para Sumarização de Documentos Textuais

Neste capítulo, são apresentadas as abordagens propostas para sumarização automática de documentos textuais. Inicialmente, na Seção 4.1, é exposta a formalização do problema de sumarização extrativa. Na Seção 4.2, são apresentadas as características dos documentos das NCs, identificados nos dados analisados. Na Seção 4.3, são apresentadas as abordagens propostas enfatizando-se as etapas do fluxo de execução e os passos de cada etapa. Por fim, na Seção 4.4, são discutidas as considerações finais do capítulo.

4.1 Formalização do Problema de Sumarização Extrativa

Esta seção apresenta a formalização do problema de sumarização extrativa. Nesta pesquisa, o problema em questão foi definido como uma tarefa classificação de sentenças. Neste sentido, dado um documento D contendo várias sentenças $(s_1, s_2, ..., s_i)$, onde s_i é a i-ésima sentença no documento, um sumarizador extrativo visa produzir um subconjunto de D para formar o resumo S selecionando m sentenças de D (onde $m < i$). A sumarização extrativa é a tarefa de atribuir uma pontuação $p(y_i|s_i, D, \theta)$ para quantificar $s_i's$ a relevância de cada sentença $s_i \in D$ em relação ao resumo e atribuir um rótulo a cada sentença $y_i \in 0, 1$ (onde 1 significa que s_i deve ser incluído no resumo; 0, caso contrário). Os parâmetros do modelo são denotados por θ. A função $p(y_i|s_i, D, \theta)$ é estimada usando um modelo, e um resumo S é criado selecionando as m sentenças com $p(1|s_i, D, \theta)$ melhores pontuações.

4.1.1 Definição de SA de texto

Dado um documento $T = (S_1, S_2, ..., S_L)$ contendo L sentenças e seu resumo de referência S^*, um modelo de sumarização extrativa deve selecionar um subconjunto de sentenças em T para formar o resumo de saída $S = \hat{S}_i | \hat{S}_i \in T$. Um resumo de candidato S é medido calculando o valor ROUGE [32] entre S e S^* usando uma pontuação ao nível de sentença:

$$g^{sen}(S) = \frac{1}{|S|} \sum_{s \in S} Rouge(s, S^*) \tag{4.1}$$

onde s é uma sentença em S e $|S|$ representa o número de sentenças em S. $Rouge(\cdot)$ denota a soma das médias das pontuações ROUGE-1, ROUGE-2 e ROUGE-L F1. Portanto, $g^{sen}(S)$ indica a sobreposição média entre cada sentença em S e o resumo de referência S^*.

Na etapa de treinamento, o resumo de referência (S^*) e o resumo de saída (S) estão disponíveis. O objetivo do treinamento é aprender uma função de pontuação $f(S)$ que possa maximizar uma função de avaliação $r(S, S^*)$ e ser utilizada para encontrar o melhor resumo durante o teste:

$$\begin{aligned} & \max_{S \in T} && f(S) \\ & s.t. && S = \hat{S}_i | \hat{S}_i \in T \\ & && |S| \leq i. \end{aligned} \tag{4.2}$$

onde i é o limite de comprimento do resumo gerado. Nesta pesquisa, i é o limite do número de sentenças que farão parte do resumo. Por fim, espera-se que o modelo de sumarização aprenda a classificar as sentenças de acordo com a importância real da sentença para o entendimento do documento.

4.2 Características dos Documentos das Notícias Crime

Ao analisar os documentos das NCs considerados nesta pesquisa, alguns características foram identificados por meio das visualizações a seguir, como:

- Os tamanhos dos documentos costumam variar entre 1 até 300 páginas;

- As sentenças mais importantes dos documentos geralmente não estão no início do documento;
- Presença de documentos de diferentes subdomínios;
- Os documentos das notícias crime geralmente possuem baixa qualidade digital (e.g. nitidez, qualidade do escaneamento) e contém formulários, tabelas e arquivos que aumentam a complexidade do documento e precisam ser removidos do texto.

Nas Figuras 4.1 e 4.2, são exibidos o número de páginas e o número sentenças no conjunto de dados das NCs. O eixo horizontal representa o número de páginas e sentenças em cada documento e o eixo vertical a densidade do número de documentos. Na Figura 4.2, observa-se que o tamanho dos documentos costuma variar bastante, existindo documentos com mais de 200 páginas. Além disso, na Figura 4.1, observa-se que existe um pico inicial de documento com poucas sentenças (menos do que 20). Por outro lado, existe outro pico com uma crescente de documentos com mais de 100 sentenças. Sendo assim, podemos inferir que existe uma variação no tamanho dos documentos. Por fim, vale ressaltar que a tarefa de sumarização se torna mais complicada à medida que o tamanho dos documentos aumenta.

As posições das sentenças mais importantes dos documentos das NCs são apresentadas na Figura 4.3. O eixo horizontal representa as posições das sentenças mais importantes dos documentos e o eixo vertical representa a densidade do número de sentenças. Nessa imagem, observa-se que existem muitas sentenças importantes que não estão no início do documento, indicando que o conjunto de dados das NCs não segue o padrão da pirâmide invertida, padrão em que as primeiras sentenças do documento contêm informações importantes, fazendo com que o modelo Lead-3 o qual captura as três primeiras sentenças supere a maioria das técnicas de sumarização. Essa característica dificulta ainda mais o processo de sumarização, pois a estratégia de sumarização não pode se basear na posição das sentenças para indicar sua importância, além de ter que analisar todo o documento.

Outra característica do conjunto de dados das NCs é a presença de subdomínios, como ilustrado na Figura 4.4. O eixo horizontal representa as áreas de atribuição (subdomínios) da Polícia Federal presentes no conjunto de dados e o eixo vertical representa a quantidade de documentos presente em cada área de atribuição. Nessa imagem, observa-se que existe um desbalanceamento das áreas de atribuição, com uma grande presença de algumas áreas

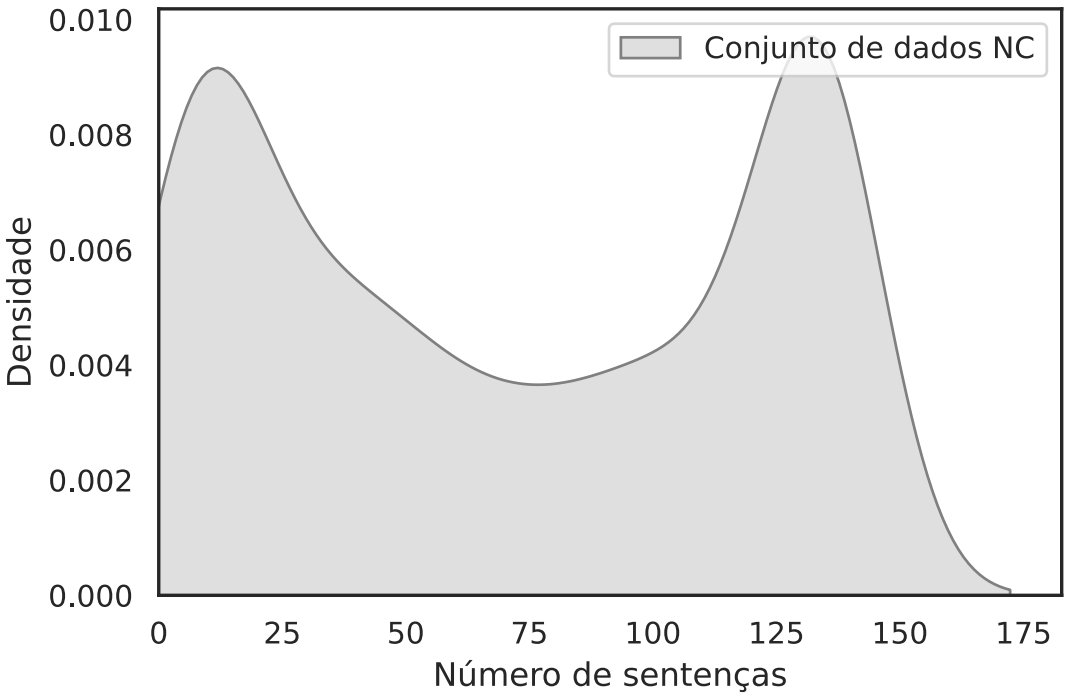


Figura 4.1: Densidade do número de sentenças nos documentos das NCs.

de atribuição e poucos exemplos de outras. Além disso, existem documentos que não estão categorizados em nenhuma das áreas existentes. Por fim, a presença de uma grande quantidade de subdomínios pode dificultar a aprendizagem dos modelos ao identificar padrões para sumarização dos documentos das NCs.

O conceito de domínio utilizado nesse trabalho refere-se ao conjunto de dados que serão utilizados como entrada para treinamento e avaliação das abordagens como, por exemplo, o conjunto de dados das NCs. Por outro lado, o conceito de subdomínio refere-se a subgrupos dentro do domínio que possuem características e padrões similares entre sí como, por exemplo, documentos de NCs que relatam crimes cibernéticos. Em relação ao domínio das NCs, existem sobreposição de documentos em subgrupos. Um exemplo seria um documento que relata um crime cibernético e previdenciário; esse documento pode pertencer a qualquer um dos dois subgrupos. Nesse trabalho, utilizamos a atribuição designada ao documento em questão pelo policial federal.

Na Figura 4.5, é ilustrada a qualidade digital de um documento de NC e duas estruturas presentes nos documentos: formulário e arquivo. A presença de páginas com baixa qualidade

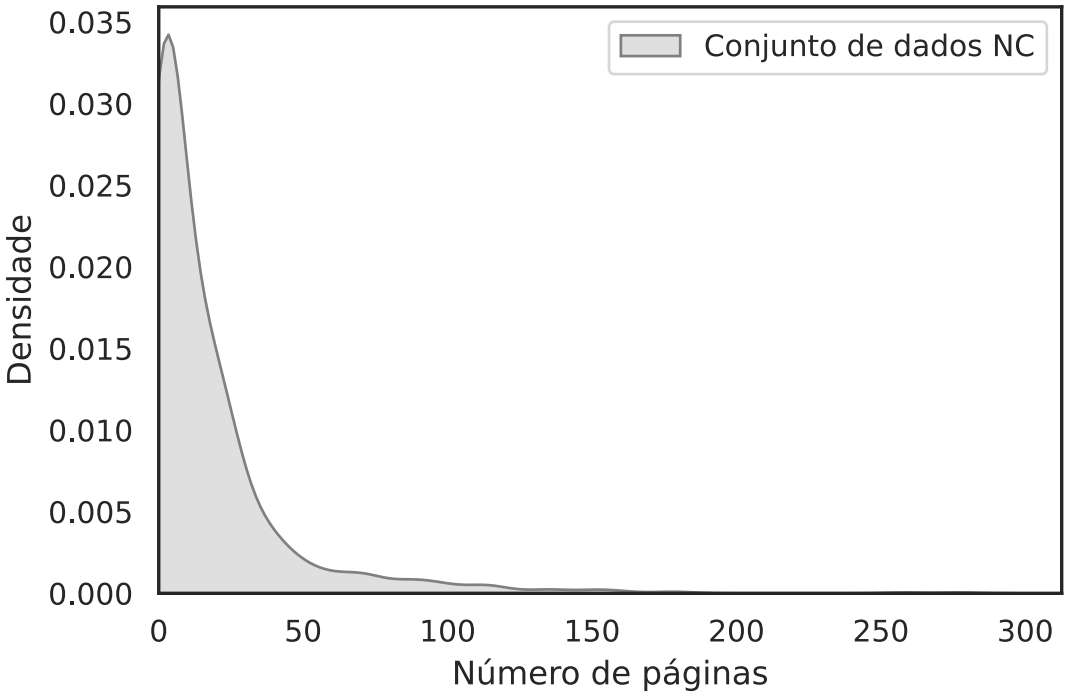


Figura 4.2: Densidade do número de páginas nos documentos das NCs.

digital e de estruturas dificultam a criação de resumos, pois partes do texto são extraídas com erros de ortografia. Além disso, parte dessas estruturas são extraídas como textos que farão parte do documento a ser sumarizado, poluindo o texto do documento e dificultando a tarefa de sumarização.

4.3 Abordagens para Sumarização Automática de Documentos Textuais

Considerando os desafios relacionados ao documentos das NCs, criar um modelo capaz de gerar resumos de qualidade nesses documentos é um tarefa complexa. Para lidar com documentos de NCs, é necessário definir um modelo de sumarização robusto que consiga lidar com documentos de diferentes subdomínios e tamanho arbitrário. Sendo assim, este trabalho apresenta três abordagens para sumarização de documentos textuais. Nesta seção, são apresentados os detalhes sobre as abordagens propostas nesta pesquisa. As abordagens foram construídas com base no modelo BERTSUM e fornecem a capacidade de lidar com

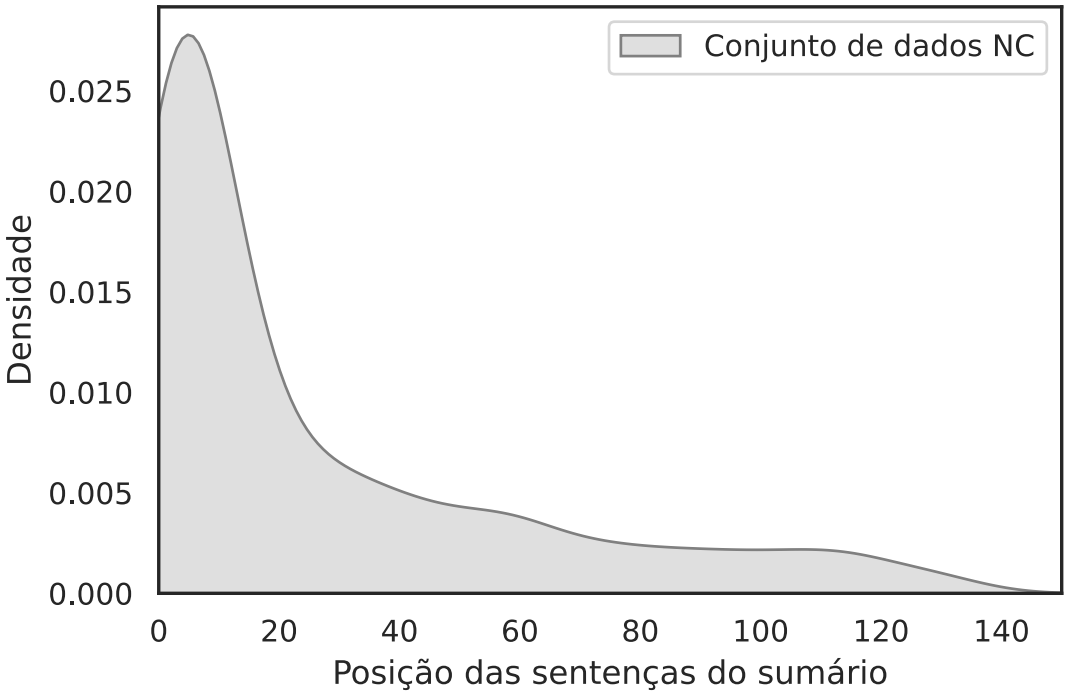


Figura 4.3: Densidade das posições das sentenças mais importantes nos documentos das NCs.

documentos de comprimento arbitrário. Vale ressaltar que uma das abordagens foi proposta para lidar com documentos de diferentes subdomínios utilizando como base uma técnica de agrupamento automático. Neste trabalho, não foi utilizado o conhecimento prévio do subdomínio de cada documento, pois novos documentos não terão essa informação. Além disso, não foram utilizadas técnicas de classificação, pois essas técnicas só se aplicariam a base de dados que possuem o rótulo do subdomínio de cada documento da base. Também é explicado, nesta seção, como as arquiteturas proposta em cada abordagem diferem da arquitetura original do BERTSUM que foi utilizada como base para criação das abordagens.

Neste trabalho, foram aplicados modelos de aprendizagem profunda para sumarização automática de documentos. Estes modelos, em oposição aos modelos estatíticos, buscam descrever as propriedades dos dados sem conhecimento prévio da distribuição dos mesmos. Por não dependerem explicitamente de parâmetros para modelar o comportamento do evento, esses modelos são mais simples de serem ajustados e demonstram considerável desempenho mesmo quando aplicados a relacionamentos complexos e não lineares como, por exemplo,

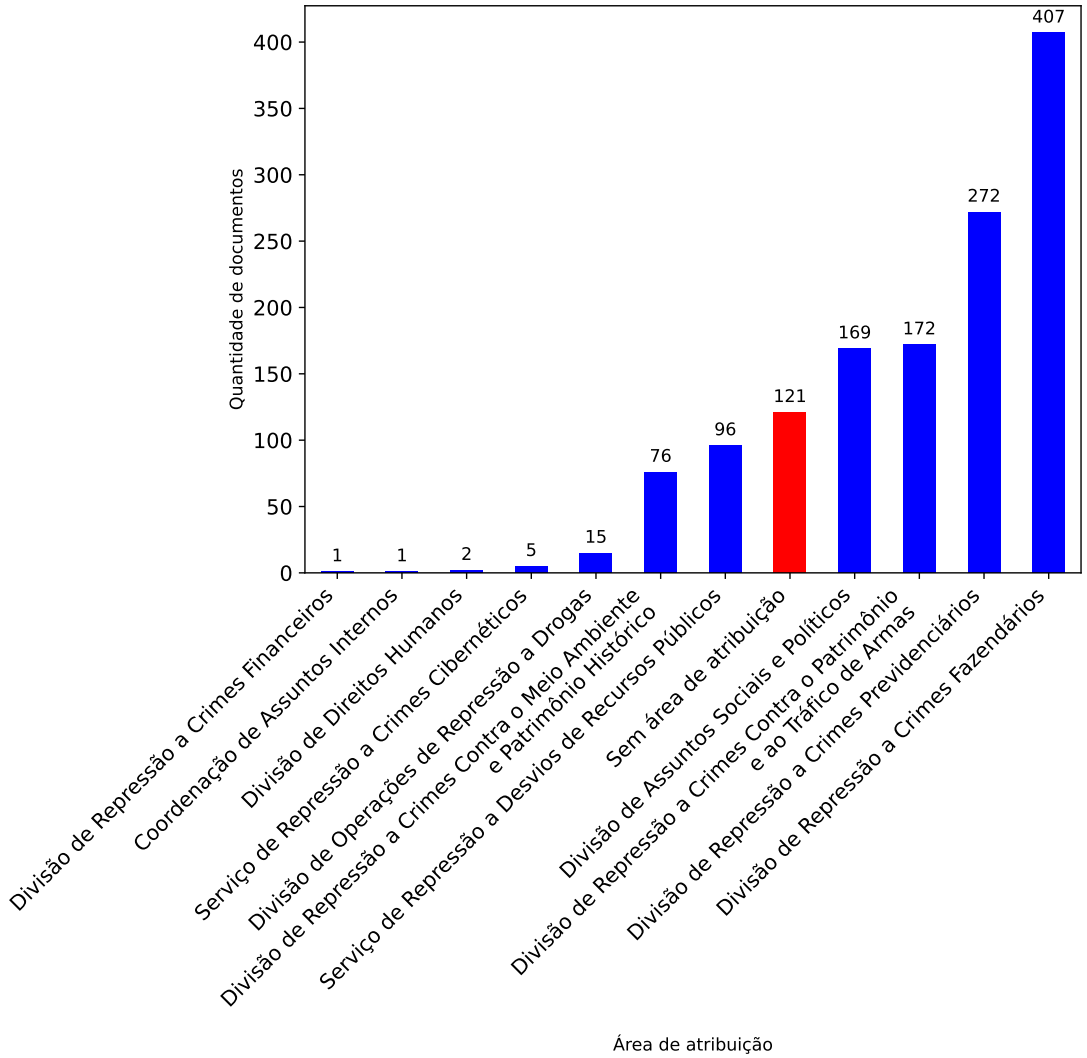


Figura 4.4: Áreas de atribuição dos documentos das NCs.

uma função de relação para identificar as sentenças mais importantes dos documentos [44]. Os modelos de aprendizagem profunda aplicados nesse trabalho têm como base um modelo pré-treinado. Os modelos pré-treinados, por terem sido treinados em um grande corpus, são mais simples, rápidos e não necessitam de uma grande quantidade de dados para serem ajustados para uma tarefa específica. Além disso, em geral, os modelos pré-treinados apresentam melhores resultados, em termos de eficácia, quando comparados a modelos treinados do zero.

Nesse sentido, foi considerado o modelo pré-treinado BERT, que representa o estado-da-arte em diversas tarefas de PLN como, por exemplo, a tarefa de classificação de texto


MINISTÉRIO DO MEIO AMBIENTE
INSTITUTO BRASILEIRO DO MEIO AMBIENTE E DOS RECURSOS NATURAIS RENOVÁVEIS - IBAMA
DIRETORIA DE PROTEÇÃO AMBIENTAL – DIPRO
COORDENAÇÃO GERAL DE FISCALIZAÇÃO AMBIENTAL – CGFIS
COORDENAÇÃO DE OPERAÇÕES DE FISCALIZAÇÃO - COFIS

RELATÓRIO DE FISCALIZAÇÃO (AUTUAÇÃO)

NOME DA OPERAÇÃO	Nº DA ORDEM DE FISCALIZAÇÃO	Nº DO AUTO DE INFRAÇÃO	DATA DO RELATÓRIO
		9091113-E	21/03/16

DADOS DO AUTUADO

Nome:	[redacted]
CNPJ/ CPF	
	12.131.131/0001-97

EQUIPE DE FISCALIZAÇÃO

NOME DO SERVIDOR	INSTITUIÇÃO	MATRÍCULA	LOTAÇÃO
[redacted]	IBAMA	[redacted]	NUCOF

CARACTERÍSTICAS DA INFRAÇÃO

Provocada	x	Negligenciada	
Ação de terceiros		Ação com a participação de terceiros	

Obs.:

IDENTIFICAÇÃO DA AUTORIA, HISTÓRICO, ABORDAGEM E CONSTATAÇÃO DA INFRAÇÃO

Em atendimento ao Processo de nº: 642/14-83,
Obs.: Informo que o referido auto de infração originou-se do termo de apreensão dos Correios de nº: S1637943914BR, onde encontra-se acostado ao referido processo Termo de apreensão do Ibama de nº:23671-E datado de 30.01.14, Termo de Doação de nº: 016090 Série B datado de 11.01.14

Lei nº 9.605/98, art. 14.

Circunstâncias		Justificativas
I - Baixa escolaridade		
II - Arrependimento eficaz do infrator		
III - Comunicação prévia do agente		
IV - Colaboração com a Fiscalização		

AGRAVANTES – Lei nº 9.605/98, art. 15.

Circunstâncias		Justificativas
I – Reincidência		
II – Cometimento da infração:		
a) para obter vantagem pecuniária		

Figura 4.5: Formulário e arquivo extraídos de um documento de NC.

[17]. A utilização do modelo BERT como base pode obter maior eficácia quando aplicado à tarefa de sumarização. Além disso, modelos que usam o BERT como base apresentam uma capacidade de generalização mais forte do que outros modelos de AM [17]. Mesmo que o modelo BERT seja complexo e de treinamento custoso, como é um modelo pré-treinado, é necessário apenas treinar uma camada de rede neural no topo do modelo. Portanto, algumas motivações dessa técnica são a diminuição do erro de generalização, a não necessidade de uma grande quantidade de dados para treinamento, robustez e comprovação da eficácia do modelo em tarefas de NLP, visando melhorar a estabilidade e eficácia das predições.

Para utilizar o modelo BERT como base para as abordagens propostas nesse trabalho, foi considerado o modelo *BERTimbau*, uma versão do modelo BERT treinada em um corpus com textos em português [63]. A variante BERT utilizada nesta pesquisa foi a BERT-large

com $L = 24$ (número de camadas de codificadores), $H = 1024$ (dimensão dos embeddings gerados) e $A = 16$ (número de camadas de atenção (self-attentions heads)). Além disso, as abordagens foram construídas sobre a arquitetura do modelo BERTSUM, porém alterando sua estrutura a fim de permitir lidar com documentos de comprimento arbitrário. Também foi proposta uma abordagem para lidar com a sumarização automática de textos multidomínio, que utiliza uma técnica de agrupamento automático.

Nas próximas seções serão descritas mais detalhadamente as abordagens propostas nesta pesquisa. Essas abordagens são: BERTSUM-ALD; BERTSUM-ALD-MD e BERTSUM-ALD-ES.

4.3.1 BERTSUM-ALD

A abordagem BERTSUM-ALD (Figura 4.6) foi construída sobre o modelo BERTSUM, mas tem um componente chave diferente: a multi-entrada, ou seja, a possibilidade de receber um número variado de subdocumentos na entrada de uma vez. Isso dá ao BERTSUM-ALD a capacidade de lidar com documentos de comprimento arbitrário.

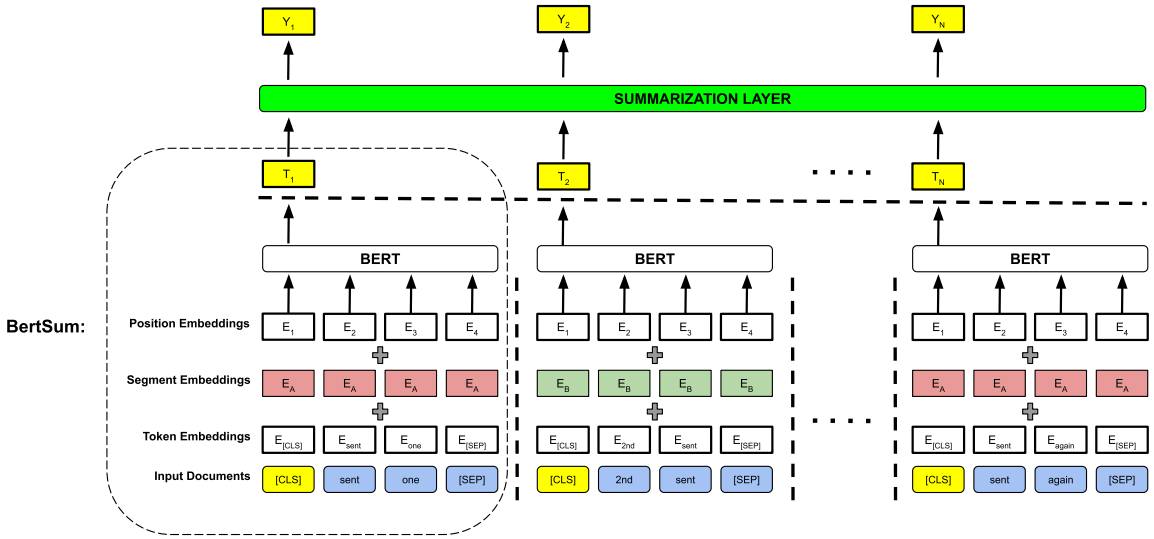


Figura 4.6: Abordagem BERTSUM para SA de documentos de tamanho arbitrário

Multi-entrada. Supondo que o comprimento do documento é m_{doc}, e temos apenas um modelo BERT pré-treinado com o comprimento máximo de sequência m_{BERT}. Primeiro dividimos o documento em n_{subdoc} subdocumentos contendo $n_{sentencas}$ e cada subdocumento

tem o comprimento máximo m_{BERT} (o comprimento do subdocumento $n_{subdoc-th}$ é menor que m_{BERT}). Definimos $m_{BERTSUM-ALD} = n_{subdoc} * m_{BERT}$ como o comprimento da sequência de BERTSUM-ALD. Então, temos $m_{BERTSUM-ALD} \geq m_{doc}$, o que faz com que o BERTSUM-ALD possa lidar com um documento arbitrariamente longo como entrada, sempre dividindo esse documento em subdocumentos de tamanho igual ou inferior à entrada do modelo BERT.

Funcionamento da abordagem. A abordagem BERTSUM-ALD assume o mesmo formato de entrada que o BERTSUM, ou seja, as tags [CLS] e [SEP] são adicionados ao início e ao final de cada sentença, respectivamente. Por outro lado, antes de alimentar o modelo BERT, a sequência é dividida em subdocumentos cujos comprimentos são menores ou iguais ao comprimento máximo de sequência do modelo BERT pré-treinado. Os *token embeddings* e os *segment embeddings* também são menores ou iguais aos do BERT original. Para os *position embeddings*, foram aplicados os *position embeddings* usados no BERT original, limitado a m_{BERT} de comprimento. No entanto, é copiada a matriz de *position embeddings* original n_{subdoc} vezes e, a partir daí, essas matrizes são concatenadas. Um exemplo ilustrativo é apresentado na Figura 4.7, onde o documento é dividido em três partes baseado no número máximo de tokens que o BERT suporta (512 tokens), os quais são processados em paralelo pelo modelo BERT. Cada subdocumento contém uma ou mais sentenças, são inseridas as tags de [CLS] e [SEP] em cada sentença e, logo em seguida, as sentenças são concatenadas para formar o subdocumento. Após isso, os subdocumento é concatenado com tags de [PADS] até chegar ao número de 512 tokens. Por fim, é utilizado o pré-processamento do BERT para extraír os *token embeddings*, *segment embeddings* e os *position embeddings* do texto, concatenar esses *embeddings* e passar esses *embeddings* para o modelo BERT processar o subdocumento e pontuar as sentenças.

Etapa de sumarização. O passo a passo de como gerar um novo resumo para abordagem BERTSUM-ALD é simple. A entrada é o documento. Inicialmente, o documento é pré-processado usando bertPreprocess (pré-processamento utilizado para dividir o documento em subdocumentos), conforme ilustrado na Figura 4.7. Os subdocumentos gerados e são utilizados como entrada para o modelo BERT (**linha 4**). Por fim, como essa abordagem possui apenas uma camada de sumarização, essa camada é utilizada para pontuar todas as sentenças do documento. As sentenças são ordenadas pelas pontuações, obtidas na classi-

ficação, em ordem decrescente e as primeiras m sentenças são selecionadas para formar o resumo.

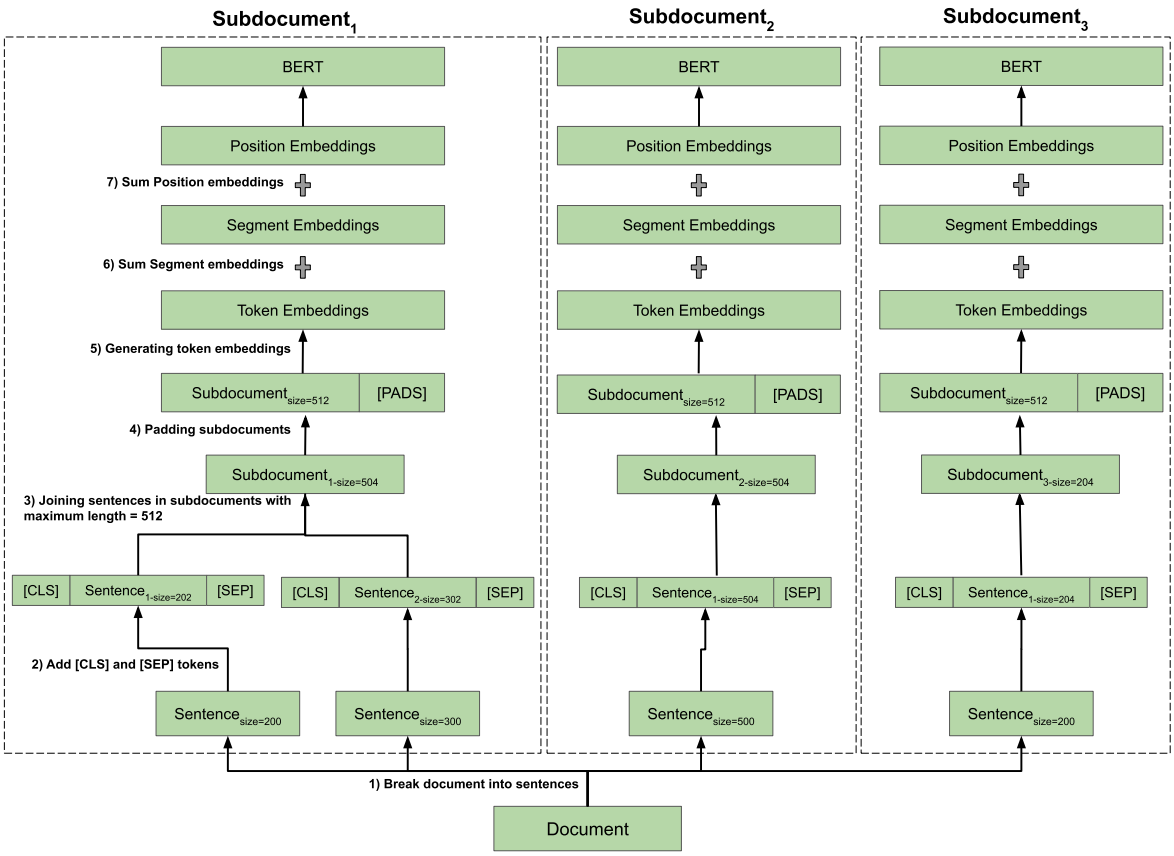


Figura 4.7: Visualização de como as abordagens lidam com documentos arbitrariamente longos.

4.3.2 BERTSUM-ALD-MD

A abordagem BERTSUM-ALD-MD (Figura 4.8) segue os mesmos princípios chave apresentados no BERTSUM-ALD de funcionamento da abordagem e da técnica de multi-endtrada para lidar com documentos arbitrariamente longos. Porém, nesta seção, é proposta uma abordagem que aproveita as técnicas de agrupamento de texto para melhorar a eficácia da sumarização automática de texto multidomínio. No topo do modelo BERT, foram implementadas n camadas de sumarização, cada uma especializada em um subdomínio específico (definido como um grupo do agrupamento). Essa abordagem se baseia na suposição de que documentos similares têm regras/padrões mais semelhantes para extrair informações do que

documentos dissimilares. Vale ressaltar que essa abordagem possui limitações devido à sua dependência do modelo de agrupamento. Em outras palavras, modelos que não agrupam documentos com boa qualidade podem impactar os resultados dos resumos produzidos por essa abordagem.

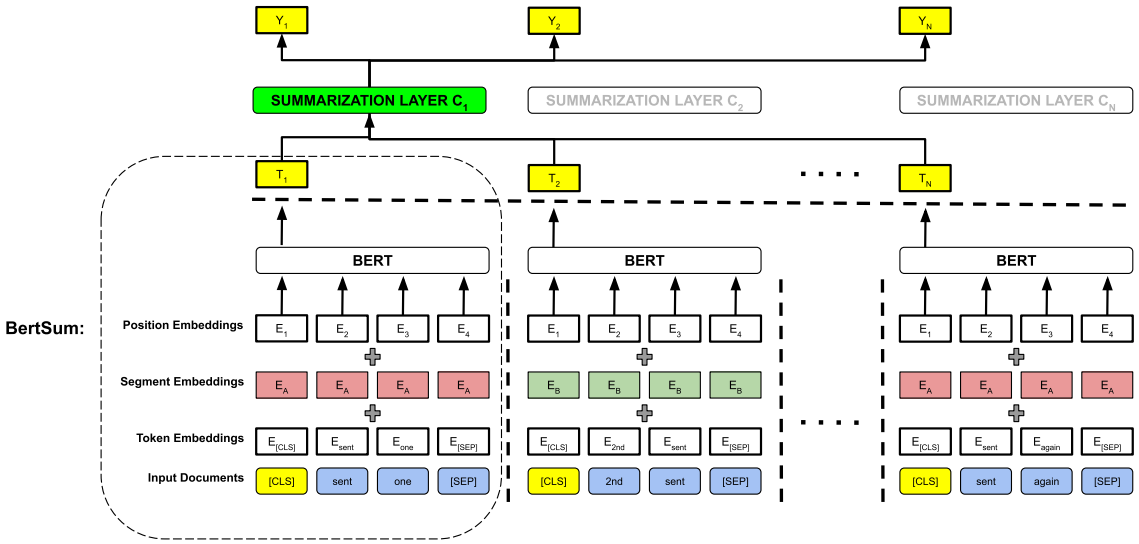


Figura 4.8: Exemplo da abordagem BERTSUM para sumarização de documentos textuais de tamanho arbitário e multidomínios

Funcionamento da abordagem. *Conforme mostrado na Figura 4.9, existem duas fases: fase de treinamento e fase de inferência. Durante a fase de treinamento, são aplicadas as etapas de pré-processamento (por exemplo, tokenização, lematização e vetorização de palavras), extração de features (características) e criação de um modelo de agrupamento para dividir os documentos da base de dados em um número* n *de subdomínios. Em seguida, são construídas e treinadas,* n *camadas de sumarização no topo do modelo BERT (dependendo do número de grupos que foram criados na etapa de agrupamento). Cada camada de sumarização é treinada utilizando os respectivos documentos atribuídos a ela. Durante a fase de inferência, são aplicadas as mesmas etapas de pré-processamento e extração de features utilizadas durante a fase de treinamento. Em seguida, é determinado o grupo ao qual o novo documento pertence e, em seguida, é utilizada a camada de sumarização treinada nesse grupo para classificar as sentenças do documento. As sentenças são ordenadas pelas pontuações, obtidas na classificação, em ordem decrescente e as primeiras* m *sentenças são*

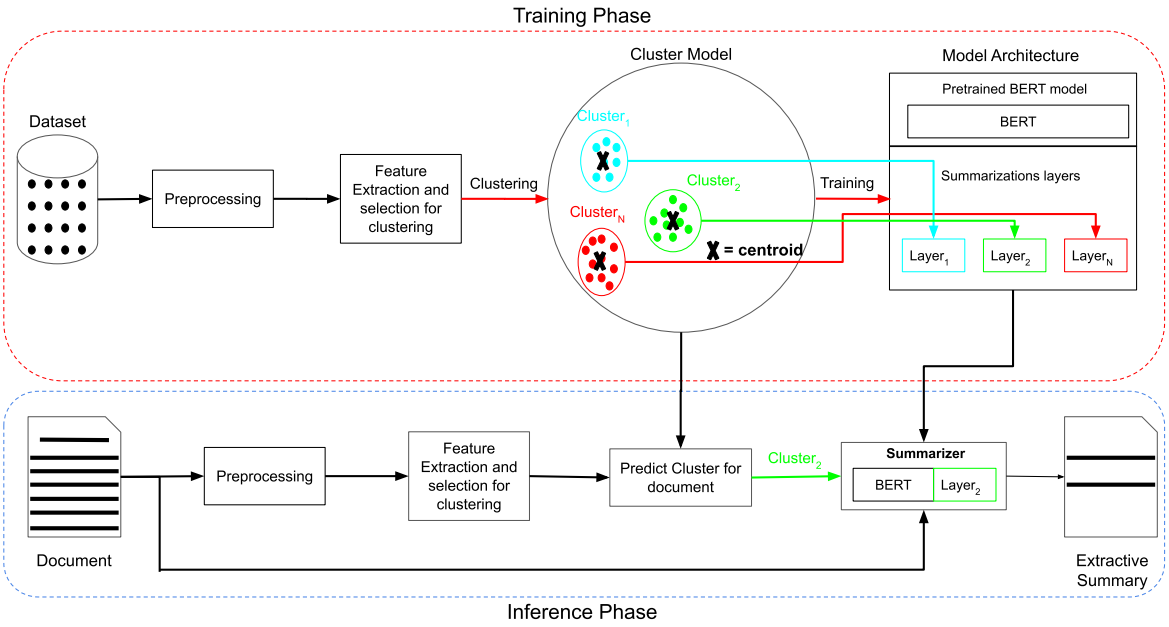


Figura 4.9: O fluxo de execução utilizando a abordagem de sumarização automática de texto multidomínio.

selecionadas para formar o resumo.

Desta forma, a abordagem BERTSUM-ALD-MD fornece uma estrutura capaz de lidar com o problema de subdomínios da base de dados das NCs. Essa abordagem aplica o agrupamento de texto para resolver o problema de dependência de domínio. Conforme mostrado na Figura 4.9, o processo começa com o treinamento da abordagem para categorizar o documento corretamente. Nesta estrutura, o processo de agrupamento de texto é aplicado primeiro. Em seguida, de acordo com o rótulo do grupo específico, uma camada de sumarização é aplicada. A estrutura pode ser combinada com qualquer técnica de agrupamento. Além disso, a abordagem também permite utilizar técnicas de classificação de texto ao invés de agrupamento de texto, considerando um contexto no qual a base de dados contenha os rótulos da categoria de cada documento.

Etapa de sumarização. O Algoritmo 4.1 fornece uma descrição passo a passo de como gerar um novo resumo. As entradas são os documentos e o modelo de agrupamento treinado para atribuir o melhor grupo (grupo com os documentos mais similares ao documento alvo) para cada documento. Inicialmente, o documento é pré-processado e as *features* são extraídas do documento (**linha 4**). Então, o algoritmo de agrupamento é aplicado para atribuir

o melhor grupo ao documento (**linha 5**). Posteriormente, na **linha 6**, o documento é pré-processado usando bertPreprocess (pré-processamento utilizado para dividir o documento em subdocumentos), conforme ilustrado na Figura 4.7. Os subdocumentos gerados e o rótulo do grupo (grupo ao qual o documento foi categorizado) são utilizados como entrada para o modelo BERT (**linha 4**). Por fim, é utilizada a camada de sumarização específica treinada naquele grupo para prever uma pontuação para cada sentença no documento. Essas sentenças são ordenadas pelas pontuações, em ordem decrescente, e as primeiras n sentenças são selecionadas para formar o resumo (**linhas 12–18**).

Código Fonte 4.1: Passo à passo da geração do sumário utilizando a abordagem de BERTSUM-ALD-MD

```
1  Entrada: D, ClusterModel: documento a ser sumarizado, modelo de
       agrupamento
2  Saida: resumo gerado
3
4  preprocessedD = featurePreprocess(D)
5  clusterD = ClusterModel.predictCluster(preprocessedD)
6  subdocumentsD = bertPreprocess(D)
7  sentencesScores = bert(subdocumentsD, clusterD)
8  sentsSorted = sort(sentencesScores)
9  numOfSentences = 0
10 summary = ""
11
12 for sentenceScoreIndex in sentsSorted:
13     senteceIndex = sentsSorted[1]
14     if numOfSentences== 0 or not trigramBlocking(summary, sentencesD[
           sentenceIndex]):
15         summary += sentencesD[sentenceIndex]
16         numOfSentences += 1
17     end if
18 end for
19
20 return summary
```

4.3.3 BERTSUM-ALD-ES

A abordagem BERTSUM-ALD-ES é similar a abordagem anterior, também seguindo os mesmos princípios chave apresentados no BERTSUM-ALD de funcionamento da abordagem e da técnica de multi-endtrada para lidar com documentos arbitrariamente longos. Além disso, essa abordagem também contém um conjunto de camadas de resumo, mas a principal diferença é que cada camada de resumo é treinada em uma amostra do conjunto de dados original. É importante destacar que as amostras são construídas retirando observações de um grande conjunto de dados, uma amostra de cada vez, e retornando-as ao conjunto de dados após as observações terem sido escolhidas. Isso permite que uma dada observação seja incluída mais de uma vez na mesma amostra.

A técnica aplicada na abordagem BERTSUM-ALD-ES é chamada de *ensemble bagging* [10], mas, em vez de criar n modelos BERT (que seriam computacionalmente caros), é utilizado apenas um modelo BERT como base e são criadas n camadas de sumarização diferentes. Sendo assim, sabendo que os modelos-base são independentes, o erro de predição do modelo diminui quando a abordagem de *ensemble* é utilizada, por meio da utilização da "sabedoria das multidões"(*wisdom of crowds*) para realizar uma predição [28]. Mesmo que o modelo *ensemble* utilize vários modelos-base internamente, ele atua e funciona como um único modelo. Por fim, a ideia é tentar reduzir o viés e a variância de tais camadas de sumarização, combinando várias delas a fim de criar uma abordagem que melhore a eficácia do modelo.

Funcionamento da abordagem. O algoritmo 4.2 descreve o passo a passo de como treinar o modelo da abordagem de *ensemble*. As entradas são os documentos, o número de camadas de sumarização que serão ajustadas e o número de documentos que serão utilizados para ajustar cada camada de sumarização. Inicialmente, uma técnica de amostragem (*bootstrapping*) é aplicada e uma amostra (*bootstrap*) é criada para cada camada de sumarização (**linhas 4–14**). Como os documentos são subamostrados com substituição, alguns documentos podem ser super-representados em muitas camadas de sumarização, enquanto outros podem estar sub-representados ou até mesmo ausentes. No caso de ausência de um documento, são aplicadas, a esse documento, todas as camadas de sumarização. Para definir quais camadas serão treinadas por cada documento, é realizada (**linhas 16-27**) uma verificação em quais amostras cada documento aparece. Posteriormente, os documentos, bem como

as camadas de sumarização atribuídas a cada documento, são usados como entrada para o *bertPreprocess* (**linha 29**). Por fim, os subdocumentos gerados são utilizados como entrada para treinar o modelo (**linha 30**). A seguir, o modelo treinado é utilizado para sumarizar novos documentos.

Código Fonte 4.2: Passo à passo do treinamento da abordagem de BERTSUM-ALD-ES

```
1  Entrada: D, Nensemble, ensembleSize: base de dados para treinamento, nú
       mero de camadas de sumarização, número de documentos para treinamento
       de cada camada de sumarização
2  Saída: S: modelo treinado
3
4  samplesLayers = dict()
5
6  for layerN in Nensemble:
7      samplesLayer = set()
8
9      for num in ensembleSize:
10         sampleNum = randomNumber(0, length(D))
11         samplesLayer.add(sampleNum)
12     end for
13     docsIndexLayers = []
14 end for
15
16 for documentNum in length(D):
17     docIndexLayers = []
18
19     for samplesLayer in samplesLayers:
20         laynerNum = samplesLayer.key()
21         samples =  samplesLayer.values()
22         if samples.contains(documentNum):
23             docIndexLayers.append(layerNum)
24         end if
25     end for
26     docsIndexLayers[documentNum] = docLayersIndex
27 end for
28
29 subdocumentsD = bertPreprocess(D, docsIndexLayers)
```

```
30 model = bert.train(subdocumentsD)
31
32 return model
```

Exemplo 1. *Considere um conjunto de dados com quatro documentos* D_1, D_2, D_3, D_4 *e três camadas de sumarização* E_1, E_3, E_3*. Primeiro, é gerada uma amostra para cada camada de sumarização:* $E_1 = D_1, D_2$*,* $E_2 = D_1, D_3$*, e* $E_3 = D_2, D_3$*. Em seguida, é realizada uma interação sobre todos os documentos e são verificadas em quais amostras cada documento aparece, então* $D_1 = E_1, E_2$*,* $D_2 = E_1, E_3$*,* $D_3 = E_2, E_3$*,* $D_4 = \emptyset$*. Como* D_4 *está ausente, são aplicadas, a* D_4*, todas as camadas de sumarização, ou seja,* $D_4 = E_1, E_2, E_3$*. Assim, na etapa de pontuação das sentenças de um documento, são aplicadas todas as camadas de sumarização atribuídas ao respectivo documento.*

Etapa de sumarização. A etapa de sumarização é similar ao Algoritmo 4.1, mas,em vez de utilizar uma técnica de agrupamento para calcular a melhor camada de sumarização a ser aplicada aquele documento, são aplicadas todas as camadas de sumarização para calcular e, logo após, somar as pontuações para obter o resultado final, conforme ilustrado na Figura 4.10.

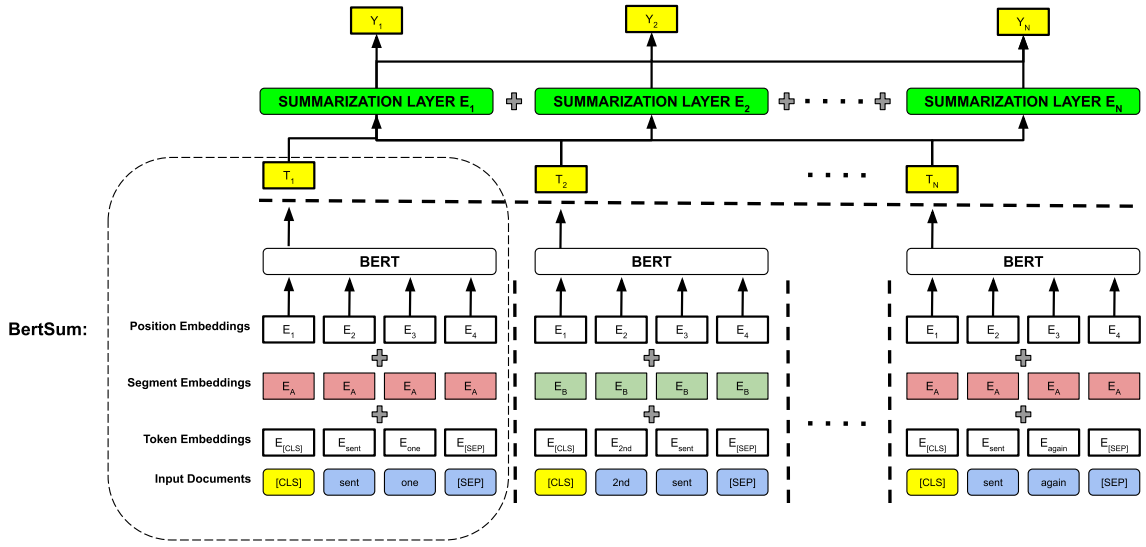


Figura 4.10: Exemplo da abordagem BERTSUM-ALD-ES para sumarização automática de documentos textuais

4.3.4 Etapa de Treinamento

Após processar os dados e adaptá-los para serem utilizados nas abordagens propostas, na etapa de treinamento as abordagens são construídas e ajustadas, como descrito a seguir.

Ajustes dos Parâmetros do Modelo

Nesta etapa, o conjunto de dados de treino é utilizado para encontrar os melhores valores dos parâmetros das abordagens. Para encontrar estes valores, foi realizado o ajuste (*tuning*) manual de parâmetros[1], escolhendo um intervalo de valores com base nos valores padrões utilizados no modelo BERT [17]. O ajuste automático apresentou-se muito custoso em termos de tempo de execução, pois um conjunto de possíveis valores é atribuído para cada parâmetro e todas as combinações resultantes do produto cartesiano desses valores são avaliadas. Visto que o treinamento de um modelo BERT pode levar em média quatro horas, não foi factível a utilização do ajuste automático. Porém, foi tunado alguns parâmetros de taxa de aprendizado e taxa de decaimento, mas não foram obtidos bons resultados.

Detalhes das Implementações das Abordagens Prospostas

Para todas as abordagens propostas, foi utilizado o *framework PyTorch* [57] e a versão brasileira do BERT [63] *neuralmind/bert-base-portuguese-cased* para implementação. Todos os documentos de entrada são tokenizados pelo tokenizador do BERT. O BERT e as camadas de sumarização são ajustados em conjunto. É aplicado no treinamento *minibatchs* com um tamanho de lote de 10 documentos em cinco épocas de treinamento. O tempo de treinamento leva cerca de quatro horas em uma única GPU. Adam com $_1 = 0,9$, $_2 = 0,999$ é utilizado para o ajuste fino, sendo essa a mesma abordagem proposta em [35]. A diferença está na taxa de aprendizagem, que é de $lr = 1e^{-4}$ (melhor taxa de aprendizagem encontranda no ajuste manual dos parâmetros). Além disso, é utilizada uma taxa de decaimento de 0,90 para estabilizar o treinamento. Após cada época, a abordagem a ser treinada é avaliada no conjunto de validação. Por fim, são selecionados os três melhores pontos de verificação com base nos erros apresentados no conjunto de validação e são relatados os resultados médios no conjunto de teste. Na fase de predição, é utilizada a abordagem treinada para obter a

[1] Os melhores valores de parâmetros encontrados estão descritos no Apêndice A

pontuação de cada sentença. Em seguida, essas sentenças são ordenadas baseadas em suas pontuações e são selecionadas as duas sentenças melhor avaliadas para gerar o resumo.

Camadas de sumarização. Como mencionado na Seção 4.3, cada abordagem treina uma ou mais camadas recebendo como entrada a saída do modelo BERT. São construídas várias camadas específicas de sumarização empilhadas nas saídas do modelo BERT para classificação das sentenças dos documentos e extração dos resumos. Essas camadas de sumarização são ajustadas em conjunto com o modelo BERT. Para a abordagem BERTSUM-ALD, é criado apenas um classificador simples. Por outro lado, na abordagem BERTSUM-ALD-MD, são criados n classificadores simples (um para cada grupo). Por fim, na abordagem BERTSUM-ALD-ES, também são criados n classificadores simples, mas o número n é definido pelo usuário. A camada de sumarização em todas as abordagens é um classificador *sigmoid*:

$$\hat{Y}_i = \sigma(W_o h_i + b_o) \tag{4.3}$$

Detalhes de implementação da abordagem BERTSUM-ALD-MD. Nesse trabalho, foi utilizada a técnica de agrupamento *K-means* para validar a abordagem proposta. Essa técnica foi escolhida pois, teve a melhor eficácia em criar grupos contendo documentos similares e grupos com tamanhos parecidos. Para isso, foi empregada a versão *Sklearn K-means* [9]. Vale ressaltar que, no caso do conjunto de dados das NCs os grupos criados pelo modelo de agrupamento diferentem das área de atribuição dos documentos, ou seja, nos grupos criados geralmente existiam documentos de diferentes área de atribuição. Além disso, para vetorizar os documentos de texto, foi aplicada a técnica de vetorização TF-IDF [2]. Usando o método do cotovelo, foi observado que o melhor quantidade de grupos para o conjunto de dados das NCs é $K = 5$. No entanto, devido às limitações do tamanho do conjunto de dados, o uso de $K = 5$ gerou grupos com um número de documentos inferior a 100, sendo esse um número insuficiente de exemplos de treino para ajustar uma camada de sumarização. Portanto, foi definido um subcojunto ótimo de $K = 3$, onde foram obtidos os melhores resultados nesse conjunto de dados. Vale ressaltar que foram avaliadas outras técnicas de agrupamento como, por exemplo, agrupamento por densidade e agrupamento hierárquico,

[2] *Term Frequency - Inverse Document Frequency* (TF-IDF) é uma técnica de vetorização de texto baseada no modelo *Bag of words* (BoW)

mas ambos não se ajustaram bem ao conjunto de dados das NCs, apresentando grupos com baixa quantidade de documentos (grupos contendo menos do que 100 documentos) e grupos contendo documentos com baixa similaridade entre sí.

Detalhes de implementação da abordagem BERTSUM-ALD-ES. Em relação à abordagem de conjunto, foram definidos números de camadas iguais a 5, 10 e 15, e números de amostras iguais a 200 e 300 no conjunto de dados das NCs. Os melhores resultados foram obtidos quando o número de camadas e amostras foi 10 e 300, respectivamente.

Bloqueio de trigramas. Durante a fase de predição, o bloqueio de trigramas é usado para reduzir a redundância. Essa etapa é necessária porque as abordagens propostas não reconhecem a redundância entre sentenças avaliadas. Então, dado um resumo selecionado S e uma sentença candidata c, c não fará parte do resumo se houver uma ou mais trigramas sobrepostas entre c e S. Essa técnica, mesmo simples, tem se mostrado eficaz para remover a redundância em pesquisas de sumarização.

4.3.5 Etapa de Teste

Por fim, na etapa de teste, as abordagens propostas são avaliadas e disponibilizadas para implantação, como descrito nos seguintes passos.

Avaliação do Modelo Proposto

Na última etapa, os dados de teste são utilizados para validar as abordagens propostas, comparando cada resumo gerado com os resumos de referência. Para avaliação, as medidas de eficácia utilizadas são ROUGE-1, ROUGE-2 e ROUGE-L.

4.4 Considerações Finais

Este capítulo apresentou a formalização do problema de sumarização extrativa, algumas características presentes nos documentos dos conjuntos de dados das NCs e, por fim, as abordagens propostas. Além disso, foi visto como a técnica de agrupamento automático pode ser utilizada para lidar com documentos de subdomínios diferentes. Por fim, as etapas do fluxo de execução do modelo e os passos de cada uma foram descritos em detalhes, bem

como a aplicação da técnica de aprendizagem incremental para atualização das abordagens propostas.

No capítulo seguinte, será apresentada a avaliação experimental das abordagens, desde a metodologia empregada até a discussão dos resultados alcançados.

Capítulo 5

Avaliação Experimental

Neste capítulo, é apresentada a avaliação experimental realizada sobre as abordagens propostas. Os experimentos foram projetados com o intuito de mensurar a eficácia e eficiência das abordagens, principalmente em termos de $F_{measure}$ e tempo de predição. O modelo proposto foi comparado com outros já empregados no problema analisado: Lead-3, TextRank e BERTSUM. Para avaliar tais modelos, foram utilizadas duas bases de dados reais (NCs e Wikihow), cujos detalhes são apresentados na Seção 5.6.

Para conduzir a avaliação experimental das abordagens propostas, os principais passos da metodologia incluíram a definição: (i) das questões de pesquisa; (ii) das métricas de avaliação; (iii) dos testes estatísticos; (iv) da técnica de ajuste de hiperparâmetros; e (v) dos dados a serem avaliados.

Na Seção 5.1, são apresentadas as questões de pesquisa que motivaram a execução dos experimentos. Na Seção 5.2, são descritas as estratégias utilizadas para coleta de dados. Na Seção 5.3, são descritas as métricas utilizadas para mensurar o modelo avaliado. Já na Seção 5.4, é definido o teste estatístico empregado nos experimentos para avaliar a significância estatística dos resultados. Na Seção 5.5, são apresentadas as técnicas de ajuste de hiperparâmetros empregadas. Na Seção 5.6, são apresentadas as bases de dados utilizadas. Na Seção 5.7, são apresentadas as etapas de pré-processamento das base de dados utilizadas no experimento. Na Seção 5.8, é apresentado o resumo dos resultados de cada experimento. Na Seção 5.9, é apresentado a discussão dos resultados obtidos no experimento. Na Seção 5.10, são discutidas as ameaças à validade das abordagens propostas. Por fim, na Seção 5.11, são apresentadas as considerações finais do capítulo.

5.1 Questões de Pesquisa

Para guiar a avaliação das abordagens propostas de sumarização automática de documentos textuais, as questões de pesquisa (QP) foram divididas em dois cenários de avaliação.

O primeiro cenário engloba questões de pesquisa relativas à **eficácia** das abordagens:

- **QP1:** A multi-entrada permite predizer com eficácia as sentenças mais importantes do documento?
- **QP2:** A abordagem de sumarização multi-domínio produz resultados superiores em relação as abordagens que não são multi-domínio?
- **QP3:** A abordagem de combinação de modelos (*ensemble*) produz resultados superiores em relação ao uso de modelos individuais?
- **QP4:** As abordagens de sumarização propostas neste trabalho produzem resultados superiores em relação aos modelos *baselines* e ao modelo utilizado no estado da arte para sumarização automática de documentos textuais?

O segundo cenário engloba uma questão relacionada à **eficiência** das abordagens, em termos de tempo para predição:

- **QP5:** As abordagens de sumarização propostas nesta dissertação permitem uma predição eficiente, em termos de tempo para predição, das sentenças mais importantes do documento?

5.2 Coleta de Dados

Nesta seção, é apresentada a metodologia aplicada para extrair os documentos dos dois conjuntos de dados utilizados na pesquisa.

- **Conjunto de dados das NCs:** inicialmente, os dados são coletados da base de dados da Polícia Federal e processados para se adequar a tarefa de sumarização extrativa. No processamento dos dados, como os documentos das NCs estão em formato PDF, foi necessário utilizar a biblioteca *Pdftotext* [56] da linguagem *Python* para extrair

os textos dos arquivos. Os documentos pertencem a diferentes áreas de atribuição e cobrem uma ampla gama de tópicos;

- **Conjunto de dados *Wikihow* brasileiro:** como os documentos estão em páginas *Web*, foi utilizada a biblioteca de *Python Scrapy* [30] para extrair os dados textuas do site *WikiHow*[1]. Os documentos estão classificados em mais de 20 categorias diferentes. Para preparar os dados para a tarefa de sumarização, cada método de "como fazer"(se houver mais de um) descrito no documento é considerado como um documento separado. Para gerar os resumos de referência, as linhas em negrito (mostradas em caixas vermelhas) que representam o resumo das etapas são extraídas e concatenadas, conforme mostrado na Figura 5.1. As descrições detalhadas de cada etapa são utilizadas para formar o documento. Vale ressaltar que esses documentos apresentaram sumários de referência com baixas qualidades de acordo com a métrica ROUGE. Por fim, vale destacar que, na ilustração presente na Figura 5.1, os documentos e os resumos estão truncados e os textos apresentados não estão no tamanho real.

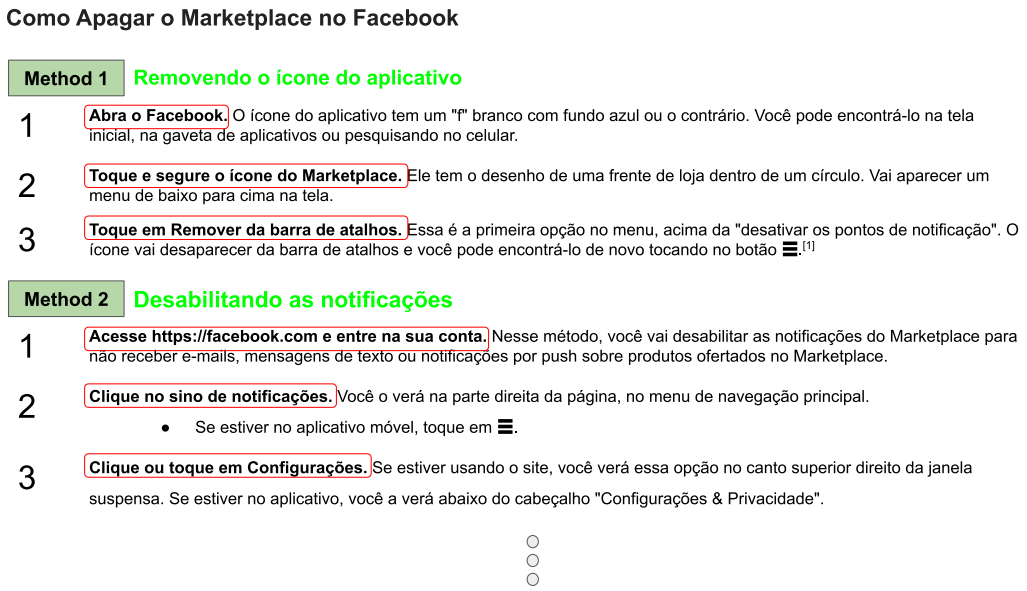


Figura 5.1: Um exemplo do nosso conjunto de dados da Wikihow: conjunto de dados da WikiHow, que inclui mais de 100 mil documentos.

[1]https://pt.wikihow.com/Página-principal

5.3 Métricas Utilizadas

Para responder às questões de pesquisa definidas na seção anterior, são utilizadas diferentes variantes da métrica ROUGE [32]. As métricas ROUGE são utilizadas para avaliar a qualidade sintática dos resumos gerados pelas abordagens propostas. Foram relatadas as pontuações de ROUGE-1, ROUGE-2 e ROUGE-L dos resumos gerados comparados com os resumos de referência, para avaliar a eficácia das diferentes abordagens, com foco na portuação $F_{ROUGE-N}$.

Embora a métrica ROUGE seja a métrica de avaliação padrão para o resumo automático de texto, ela tem limitações devido à sua natureza sintática, que não captura relacionamentos semânticos entre os textos.

Nesta pesquisa, ao lidar com um problema de classes desbalanceadas, a principal métrica de eficácia a ser considerada é a $F_{measure}$, pois representa a combinação de quanto o modelo acertou em termos de precisão e cobertura.

5.4 Testes Estatísticos

Para avaliar a significância dos resultados experimentais desta pesquisa, dois testes estatísticos não-paramétricos foram aplicados: o Teste de *Mann-Whitney* e o Teste de *Friedman*. O teste não-paramétrico não assume uma distribuição específica dos dados [14]; assim, são indicados em cenários em que a distribuição das amostras de dados é desconhecida, como é o caso das distribuição de eficácia dos modelos.

Os testes foram empregado sde acordo com as características de paridade das amostras de dados utilizadas em cada experimento. A paridade diz respeito à distribuição dos dados. Amostras pareadas são aquelas que foram extraídas da mesma população [14]. Por sua vez, amostras não-pareadas são aquelas independentes, que não são relacionadas.

Basicamente, para cada cenário de avaliação dos experimentos conduzidos, a hipótese nula diz respeito ao fato de não haver diferença estatística significativa nos resultados analisados, ou seja, na distribuição das predições avaliadas. Por sua vez, a hipótese alternativa refere-se à diferença significativa nos resultados analisados; com indícios de diferença na distribuição das predições. Nos experimentos conduzidos, a hipótese nula é rejeitada quando

o $p\_value < 0,05$, aceitando-se, portanto, a hipótese alternativa.

5.4.1 Teste de Mann-Whitney

O Teste de Mann-Whitney [39] é um teste não-paramétrico de significância estatística para determinar se duas amostras independentes foram retiradas de uma população com a mesma distribuição. Mais especificamente, o teste determina se é igualmente provável que qualquer valor selecionado aleatoriamente de uma amostra seja maior ou menor do que um valor na outra distribuição. Se violado, sugere distribuições diferentes.

5.4.2 Teste de Friedman

Para avaliar se mais de duas amostras diferentes têm a mesma distribuição ou não, o Teste de Friedman [19] pode ser utilizado. Tal teste, nomeado em homenagem à Milton Friedman, é não-paramétrico e considera que as amostras são pareadas. A hipótese nula é que as várias amostras pareadas têm a mesma distribuição. Uma rejeição da hipótese nula indica que uma ou mais das amostras tem uma distribuição diferente.

5.5 Ajuste dos Hiperparâmetros

O ajuste de hiperparâmetros refere-se ao processo de otimização automática ou manual dos valores dos hiperparâmetros de um modelo de ML. Os hiperparâmetros referem-se a todos os parâmetros de um modelo que não são atualizados durante a fase de treinamento e são usados para configurar o modelo [7]. Este processo de otimização define a combinação dos melhores valores (dentro do conjunto avaliado) dos hiperparâmetros a serem utilizados no modelo.

Algumas técnicas comumente utilizadas para realizar este processo são a Busca Aleatória (*Random Search*), Busca Manual (*Manual Search*) e Busca em Grade (*Grid Search*) [7]. No primeiro caso, uma grade dos valores dos hiperparâmetros é criada e então combinações aleatórias são extraídas. O segundo refere-se à busca manual de alguns hiperparâmetros do modelo com base no conhecimento do domínio ou do modelo. Em seguida, o modelo é treinado e avaliado com os valores escolhidos. Este processo repete-se até que uma eficácia

satisfatória seja obtida. Por fim, na técnica *Grid Search*, uma grade dos valores dos hiperparâmetros também é criada e todas as possíveis combinações de valores são avaliadas no modelo. A opções de otimização de hiperparâmetros de forma automática apresentam a desvantagem do elevado custo de processamento, em relação ao tempo de execução e consumo de memória. Nesta pesquisa, foi utilizada apenas a Busca Manual.

5.6 Bases de Dados

Neste trabalho, são utilizadas duas bases de dados, a base de dados das NCs (um conjunto de dados de domínio privado) e a WikiHow (um conjunto de dados de domínio público).

5.6.1 Base de Dados de NCs

A base de dados das NCs contém 1.000 documentos da Polícia Federal e resumos associados, ou seja, um pequeno resumo abstrativo contendo uma breve visão geral do documento, dividido em mais de 15 subdomínios (neste contexto, os subdomínios são áreas de atribuição), conforme ilustrado na Figura 4.4, onde os documentos da base de dados de NCs são divididos em subdomínios e existem documentos que não pertencem a nenhum deles.

5.6.2 Base de Dados da Wikihow

A base de dados da Wikihow brasileira é um conjunto de dados muito maior do que a NC, contendo mais de 110 mil documentos e resumos associados, divididos em mais de 20 categorias diferentes como, por exemplo, esportes, saúde, viagens e animais. Os dados foram extraídos aplicando a mesma abordagem presente em [29], cujos os documentos são extraídos das páginas através de uma técnica de *scraping* e, logo após, os textos extraídos são separados em documentos. Por fim, o resumos de cada passo são extraídos para gerar o resumo de referência e a descrições de cada passo são extráidas para gerar o documento da página.

5.7 Etapas de pré-processamento

Na primeira etapa de pré-processamento, ocorre o pré-processamento dos dados, que consiste na extração, limpeza, preparação, rotulação dos dados e divisão dos conjuntos de dados em treino e teste. Estes passos da etapa de pré-processamento são descritos a seguir.

5.7.1 Limpeza do Dados

Cada conjunto de dados coletado é analisado a fim de remover as instâncias que apresentam os seguintes valores discrepantes ou atípicos: documentos ou sumários ausentes e documentos muito curtos (ou seja, com três ou menos sentenças no documento). A remoção desses dados foi realizada, pois otimiza o tempo de processamento e treinamento dos modelos. No conjunto de dados das NCs foram removidas apenas 3 instâncias dos mil documentos. No cado do conjunto de dados da Wikihow foram removidas 2 mil instâncias das 113 mil. Além disso, documentos e sumários são necessários para que os modelos possam aprender a identificar o que é importante pois, com a ausência de documentos ou sumários, não é possível realizar o treinamento.

5.7.2 Preparação dos Dados

Ambos os conjuntos de dados contêm resumos de referência abstrativos, que não são adequados para treinar modelos de sumarização extrativa. Para adaptar os conjuntos de dados para se adequar à tarefa de sumarização extrativa, cada documento foi dividido em sentenças usando a biblioteca python NLTK [8]. Em seguida, uma técnica de algoritmo guloso (*Greedy Search*) foi utilizada para gerar um resumo oráculo (resumo que contém o conjunto de sentenças que maximizam a métrica avaliada) para cada documento. O algoritmo seleciona um conjunto de sentenças que podem maximizar as métricas ROUGE-1, ROUGE-2 e ROUGE-L F1, como as sentenças do resumo oráculo.

5.7.3 Rotulação dos Dados

Após a criação dos resumos oráculo para cada um dos documentos dos conjuntos de dados, cada sentença presente nos resumos oráculos corresponde a uma sentença que faz parte dos

respectivos documentos. Então, para rotular cada sentença nos documentos, ou seja, classificar se a sentença deve ou não fazer parte do resumo, é atribuído o rótulo 1 às sentenças selecionadas no resumo do oráculo; 0, caso contrário. Com os dados rotulados, é possível aplicar aprendizagem supervisionada no treinamento dos modelos utilizados.

5.7.4 Preparação dos dados para o modelo BERT

Para usar o modelo BERT para sumarização extrativa, é necessário aplicar um processo semelhante ao apresentado em [35], onde um *token* [CLS] (*token* de classificação) é inserido antes de cada frase e um *token* [SEP] (*token* de separação) após cada sentença. O [CLS] é *token* especial utilizado pelo modelo BERT para incorporar o significado semântico da sentença que está localizada logo após o *token*. Também foram utilizados os mesmos *segment embeddings* para distinguir várias sentenças em um documento. O vetor T_i, que é o vetor do i-ésimo *token* [CLS] da última camada do modelo BERT, é utilizado como representação da sentença $sent_i$.

Após a preparação do texto, o mesmo é dividido em n_{subdoc} subdocumentos contendo $n_{sentencas}$ e cada subdocumento tem o comprimento máximo m_{BERT}. Assim, é possível processar cada subdocumento e obter os vetores T *tokens* [CLS] para cada sentença do texto.

5.7.5 Particionamento dos Conjuntos de Dados

Por fim, cada conjunto de dados é subdividido em duas partes: 90% para treinamento e validação, e 10% para testar o modelo. A proporção da divisão não é igual porque é necessário disponibilizar para o treinamento do modelo a maior quantidade de dados possível. Em seguida, os conjunto de treinamento e validação foi divido utilizando uma técnica de *cross-validation* (validação cruzada) [2]. É um método comumente utilizado pois é simples de entender e geralmente resulta em uma estimativa menos tendenciosa ou menos otimista da qualidade do modelo do que outros métodos, como uma simples divisão de treinamento/validação. A ideia da validação cruzada é o particionamento do conjunto de dados em subconjuntos mutuamente exclusivos, e posteriormente, o uso de alguns destes subconjuntos

[2]Validação cruzada é um procedimento de reamostragem usado para avaliar modelos de aprendizado de máquina em uma amostra de dados limitada.

para a estimação dos parâmetros do modelo (dados de treinamento), sendo os subconjuntos restantes (dados de validação) empregados na validação do modelo. O parâmetro k define o número de subconjuntos a serem criados. Nesta pesquisa foi utilizado $k = 5$, sendo 4 (80%) dos subconjuntos para treinamento do modelo e 1 (20%) dos subconjuntos para validação. Esta proporção é comumente utilizada, mas não é padrão, podendo variar de acordo com a necessidade de cada problema.

5.8 Experimentos

Nesta seção, são apresentados os experimentos utilizados para avaliar as abordagens propostas, considerando as perspectivas de qualidade dos resumos gerados em termo de eficácia e tempo de predição em comparação com outros modelos *baselines*. Essas perspectivas foram agrupadas em dois cenários: avaliação de eficácia e avaliação de eficiência. Em cada cenário de avaliação, serão exibidos e discutidos os resultados obtidos por meio do ajuste dos hiperparâmetros com busca manual. Os experimentos foram realizados em um computador com Sistema Operacional Ubuntu 18.04 64-bits, Processador Intel(R) Core(TM) i7-8700 CPU @ 3.20GHz, 24GB de memória RAM, 12 CPUs e 1TB de memória auxiliar.

5.8.1 Avaliação da Eficácia

Em relação à eficácia, as abordagens propostas foram avaliadas utilizando as duas bases de dados e a métrica ROUGE. As abordagens e modelos utilizados nos experimentos foram avaliados com base na geração de resumos contendo no máximo duas sentenças. As abordagens propostas foram comparadas com outros modelos como o TextRank [40].

Abordagem multi-entrada

Alguns autores [70; 13] abordam a questão de sumarização de documentos de comprimento arbitrário com o modelo BERT e demonstram que lidar com todo o conteúdo do documento aprimora os resultados dos modelos em termos de eficácia. Por exemplo, o modelo proposto em [70] apresenta desempenho superior ao modelo estado-da-arte em sumarização extrativa. Neste trabalho, as abordagens propostas tem o intuito de serem aplicadas em documentos de comprimento arbitrário.

Nesse experimento, a abordagem multi-entrada foi aplicada em todos os documentos dos dois conjuntos de dados. Os resultados são exibidos na Figura 5.2. No gráfico desta figura e demais figuras apresentadas nesta seção, o eixo horizontal representa as medidas de eficácia e o eixo vertical representa os valores dessas medidas em termos de $F_{measure}$. Especialmente na Figura 5.2, a luminosidade da cor representa os conjuntos de dados analisados.

Em relação à QP1, a abordagem multi-entrada permite predizer com $F_{measure} \approx 80\%$ as unigramas (ROUGE-1) e subsequências comums mais longas (ROUGE-L), mais importantes nos documentos de NCs. Em relação às bigramas (ROUGE-2), os resultados foram de $F_{measure} \approx 70\%$, que pode ser considerado um bom resultado para essa métrica de sumarização. Esses resultados indicam considerável confiabilidade nas predições do modelo proposto no conjunto de dados de NCs. Além disso, a abordagem foi avaliada no conjunto de dados da Wikihow e apresentou eficácia de predição igual à $F_{measure} \approx 68\%$ para ROUGE-1 e ROUGE-L, e $F_{measure} \approx 59\%$ para ROUGE-2. Esses resultados indicam um confiabilidade um pouco inferior da abordagem multi-entrada no conjunto de dados da Wikihow. Isso se deve ao fato de que o conjunto de dados apresenta resumos de referência com baixa qualidade. Por fim, a abordagem proposta foi avaliada considerando mais de uma base de dados, sendo os valores de eficácia no conjunto de dados da Wikihow um pouco inferiores, mas indicando considerável adaptabilidade da abordagem para diferentes conjuntos de dados.

Os resultados de precisão, cobertura e $F_{measure}$ nos conjuntos de dados de NCs e Wikihow são apresentados nas Figuras 5.3 e 5.4, respectivamente. Na Figura 5.3, é possível perceber que a abordagem apresentou altos valores de cobertura ($cobertura \geq 80\%$), indicando que a abordagem consegue capturar com alta confiabilidade as partes mais importantes dos documentos. Por outro lado, a abordagem apresentou valores de precisão inferiores aos de cobertura, indicando que a abordagem gera resumos contendo informações não importantes nos documentos. Em relação ao conjunto de dados da Wikihow, as métricas apresentaram resultados similares de aproximadamente 70% para o ROUGE-1 e ROUGE-L, e 60% para o ROUGE-2. Esses valores foram aproximadamente 10% inferiores aos obtidos usando o conjunto de dados de NCs. O teste de Mann-Whitney, aplicado nas predições do conjunto de dados, apresentou $estatstica = 1852500,000$ e $p - value = 0,0$, aceitando, assim, a hipótese alternativa de que há provável diferença significativa nesses resultados.

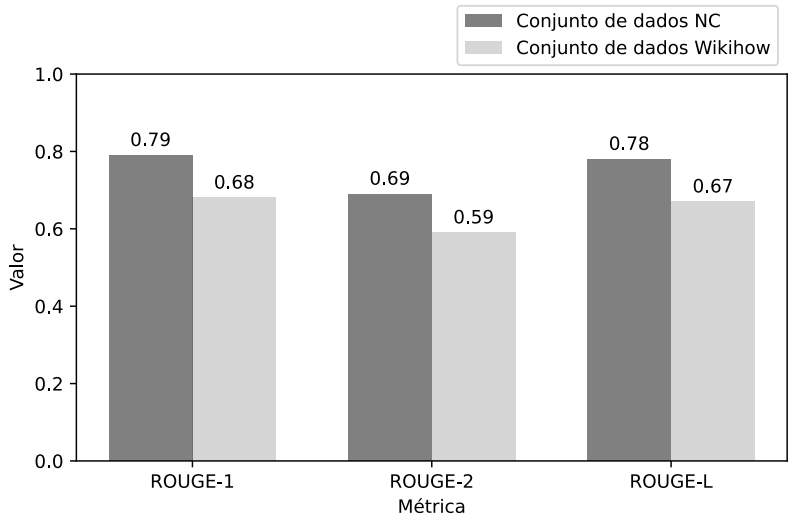


Figura 5.2: Avaliação da eficácia da abordagem BERTSUM-ALD nos conjuntos de dados NC e Wikihow

Abordagem multi-domínio

Em relação à QP2, a Figura 5.5 apresenta os resultados da abordagem multi-domínio, em termos de $F_{measure}$ nos dois conjuntos de dados. A ideia de usar uma abordagem multi-domínio é aumentar o poder de generalização do modelo, por meio da utilização de n camadas, cada uma especializada em um subdomínio específico do conjunto de dados. Assim, a abordagem multi-domínio apresentou resultados superiores aos da abordagem que utiliza apenas a multi-entrada, com $F_{measure} \geq 80\%$ para as métricas ROUGE-1 e ROUGE-L, e $F_{measure} \geq 70\%$ para a métrica ROUGE-2. Semelhante a abordagem anterior, a abordagem multi-domínio obteve resultados inferiores no conjunto de dados da Wikihow, quando comparado ao conjunto de dados de NCs, com valores de ROUGE-1 e ROUGE-L iguais à $F_{measure} \approx 70\%$, e ROUGE-L igual à $F_{measure} \approx 60\%$. Esses resultados indicam uma confiabilidade maior da abordagem multi-domínio, quando comparada com a abordagem apenas com multi-entrada. Por fim, a utilização da abordagem multi-domínio conseguiu superar os valores de ROUGE-1, ROUGE-2 e ROUGE-L da abordagem multi-entrada, nos dois conjuntos de dados avaliados.

Os resultados de precisão, cobertura e $F_{measure}$ nos conjuntos de dados de NCs e Wikihow são apresentados nas Figuras 5.6 e 5.7, respectivamente. Na Figura 5.6, é possível perceber que a abordagem multi-domínio apresentou altos valores de cobertura, chegando

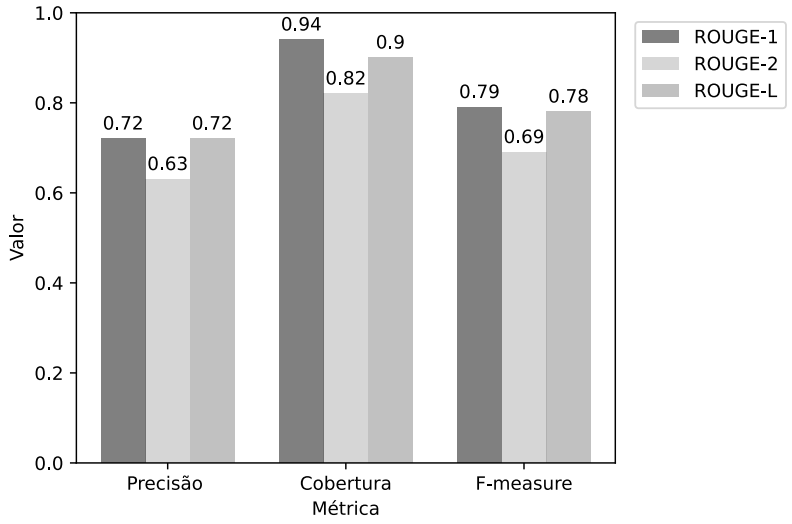


Figura 5.3: Avaliação da precisão, cobertura e $F_{measure}$ da abordagem BERTSUM-ALD no conjunto de dados de NCs

aproximadamente ao valor de 1 (cobertura perfeita) para a métrica ROUGE-1. Esses resultados demonstram alta confiabilidade da abordagem em capturar as partes mais importantes dos documentos, porém trazendo algumas partes dos documentos que não são importantes. Em relação ao conjunto de dados da Wikihow, as métricas apresentaram resultados similares de aproximadamente 71% para o ROUGE-1 e ROUGE-L e 62% para o ROUGE-2. Novamente, essa abordagem apresentou resultados superiores aos da abordagem multi-entrada. O teste de Mann-Whitney, aplicado nas predições do conjunto de dados, apresentou $estatstica = 1915000,000$ e $p - value = 0,0$, aceitando, assim, a hipótese alternativa de que há provável diferença significativa nesses resultados.

Abordagem *ensemble*

Em relação à QP3, na Figura 5.8, estão presentes os resultados da abordagem *ensemble* nos dois conjuntos de dados. A abordagem *ensemble* apresentou resultados similares à abordagem multi-domínio no conjunto de dados de NCs, com valores de $F_{measure} \approx 80\%$ para as métricas ROUGE-1 e ROUGE-L e $F_{measure} = 71\%$ para a métrica ROUGE-2. A ideia de usar uma combinação de modelos baseia-se em complementar o erro de um modelo com o acerto de outro modelo. Assim, o *ensemble*, junto com a abordagem multi-domínio, obtiveram os melhores resultados no conjunto de dados de NCs. A abordagem *ensemble*

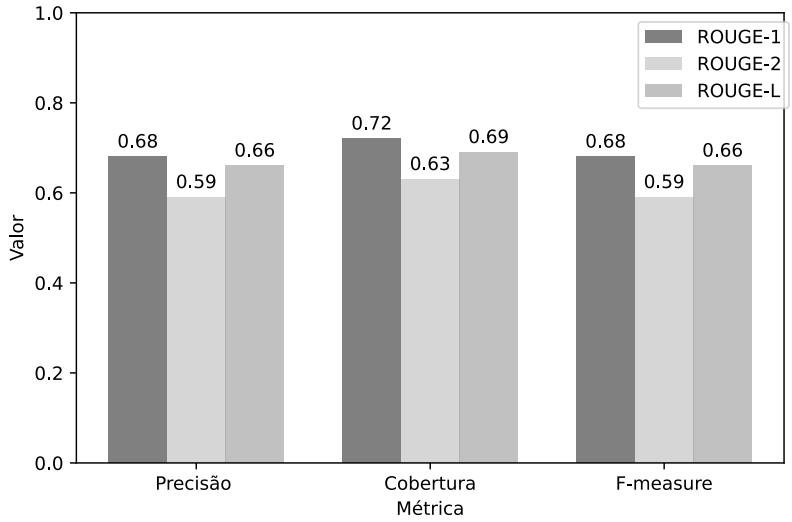


Figura 5.4: Avaliação da precisão, cobertura e $F_{measure}$ da abordagem BERTSUM-ALD no conjunto de dados da Wikihow

apresentou os melhores resultados na base de dados da Wikihow, superando a abordagem multi-domínio, com valores de $F_{measure} \approx 72\%$ para as métricas ROUGE-1 e ROUGE-L, e $F_{measure} = 64\%$, o que demonstra uma melhor adaptabilidade da abordagem quando comparada as outras abordagens propostas neste artigo, para diferentes conjuntos de dados.

Os resultados de precisão, cobertura e $F_{measure}$ nos conjuntos de dados de NCs e Wikihow foram similares aos da abordagem multi-domínio, conforme apresentados nas Figuras 5.9 e 5.10, respectivamente. A abordagem também apresentou altos valores de cobertura, porém com valores inferiores de precisão. O teste de Mann-Whitney, aplicado nas predições do conjunto de dados, apresentou $estatstica = 1585000,000$ e $p - value = 0,0$, aceitando, assim, a hipótese alternativa de que há provável diferença significativa nesses resultados.

Comparação com *Baselines*

Neste trabalho, as abordagens propostas foram comparadas com três outros modelos *baselines* que usam diferentes técnicas de sumarização: Lead-3, TextRank e BERTSUM.

Resultados no conjunto de dados das NCs. Na Figura 5.11, são apresentados os resultados das abordagens propostas comparadas com os modelos *baselines*, em termos ROUGE-1. Percebe-se que as abordagens tem resultados superiores aos outros modelos, mostrando a eficácia das mesmas.

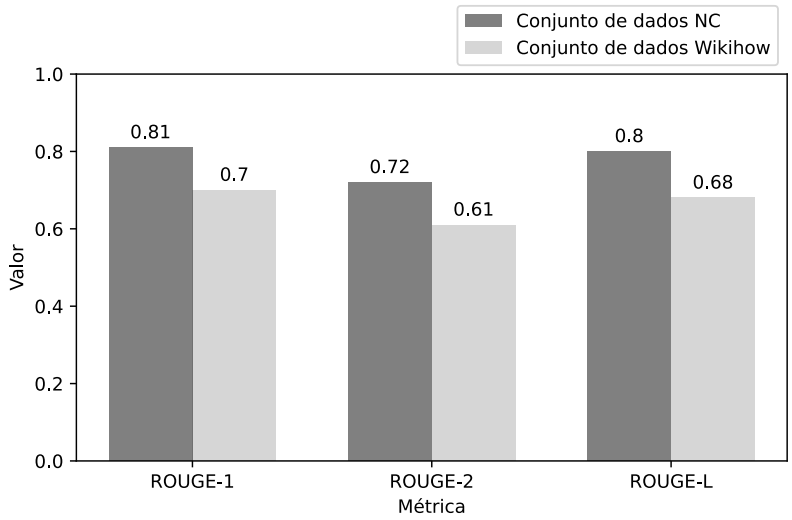


Figura 5.5: Avaliação da eficácia da abordagem BERTSUM-ALD-MD nos conjuntos de dados NC e Wikihow

A Tabela 5.1 resume os resultados de uma variedade de modelos para todas as instâncias de teste. Na tabela, observa-se que a abordagem BERTSUM-ALD-MD obtém os melhores resultados para todas as métricas ROUGE. No primeiro segmento, são apresentados os resultados da abordagem ORACLE, que é a melhor pontuação para um modelo de sumarização extrativa. Essa abordagem serve de referência para saber o máximo valor que um modelo extrativo pode alcançar de métrica ROUGE no conjunto de dados.

O segundo segmento contém os resultados dos modelos Lead-3, TextRank e BERTSUM. Embora o modelo TextRank consiga lidar com documentos de comprimentos arbitrários, seu resultado em termos de eficácia é similar aos resultados obtidos com os modelos Lead-3 e BERTSUM. Isso pode ser explicado pelo fato de o TextRank ser um modelo não supervisionado com suposições ingênuas e os documentos do conjunto de dados de NCs serem muito complexos. O Lead-3 também obteve pontuações mais baixas, o que demonstra que o conjunto de dados de NCs não segue o estilo de escrita da Pirâmide Invertida (as partes mais relevantes e importantes de um texto estão nos primeiros parágrafos) [58], comumente presente em documentos de notícias. Em relação ao modelo BERTSUM, percebe-se que o modelo apresenta um desempenho superior, por uma grande margem, em relação os outros *baselines*. Por outro lado, obteve resultados bem inferiores quando comparados aos das abordagens propostas neste trabalho. Isso se deve à limitação do modelo BERTSUM, que

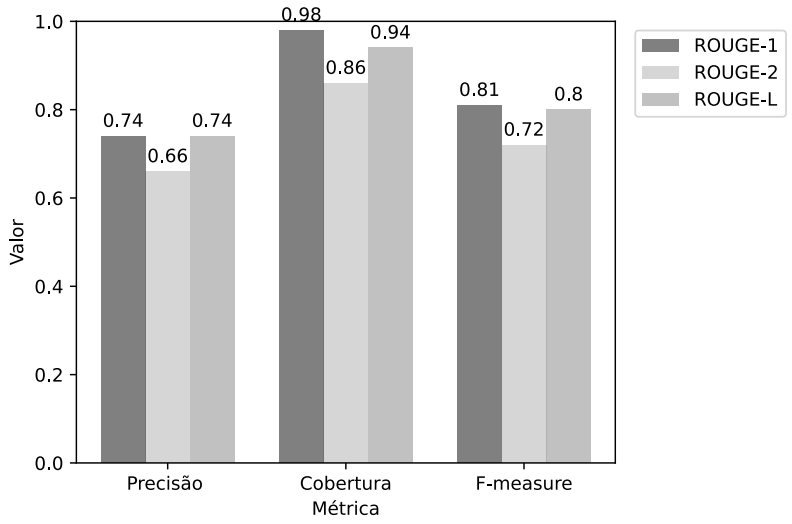


Figura 5.6: Avaliação da precisão, cobertura e $F_{measure}$ da abordagem BERTSUM-AL-MD no conjunto de dados de NCs

processa apenas os primeiros 512 *tokens* de cada documento.

Em comparação com os modelos *baselines*, as abordagens propostas superaram todos os concorrentes por uma grande margem. Por exemplo, a abordagem BERTSUM-ALD supera TextRank por 13,62 pontos na métrica ROUGE-1. Além disso, mesmo em comparação com o modelo estado-da-arte (BERTSUM), a abordagem BERTSUM-ALD (que é a abordagem mais simples) apresenta um desempenho muito superior, principalmente devido à capacidade de lidar com documentos de comprimento arbitrário, sugerindo que as sentenças mais importantes dos documentos não estão presentes apenas no início dos mesmos. Além disso, a abordagem BERTSUM-ALD-MD, que utiliza uma técnica de agrupamento, obteve uma eficácia superior à abordagem BERTSUM-ALD. Neste caso, as melhorias se devem ao fato de que a abordagem pode ajustar uma camada de sumarização específica para cada subdomínio, o que proporciona uma capacidade superior de especialização e generalização.

A abordagem BERTSUM-ALD-ES também obteve pontuações altas. Isso se deve ao fato da natureza dos modelos de *ensemble* que permite que o modelo generalize melhor, reduzindo a variância. O desempenho superior, neste conjunto de dados, demonstra a eficácia da abordagem multi-domínio. Todas as três abordagens que são variantes do modelo BERTSUM obtiveram melhorias nas métricas ROUGE. Dentre as abordagens, aquela com a técnica de agrupamento obteve o melhor desempenho. Na Figura 5.13, as curvas de densi-

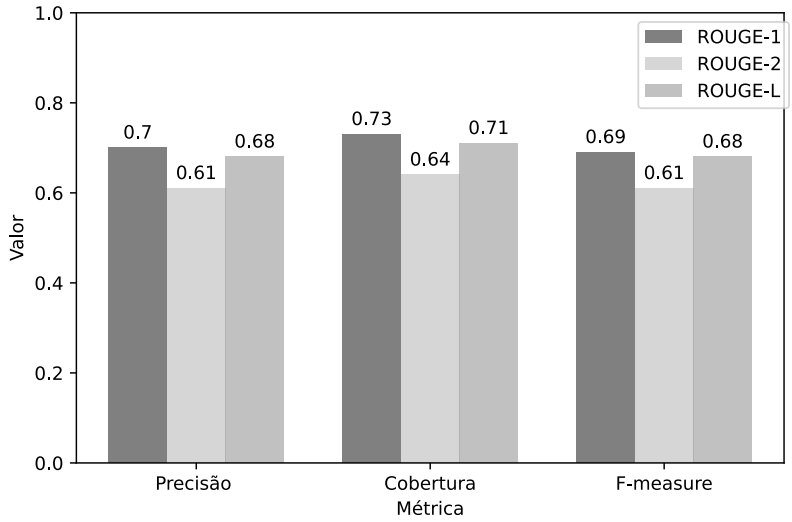


Figura 5.7: Avaliação da precisão, cobertura e $F_{measure}$ da abordagem BERTSUM-ALD-MD no conjunto de dados da Wikihow

Modelo	**ROUGE-1**	**ROUGE-2**	**ROUGE-L**
ORACLE	43.14	29.93	40.83
LEAD3	20.76	7.12	19.07
TEXTRANK	20.25	6.75	19.07
BERTSUM	29.56	15.98	27.95
BERTSUM-ALD	33.87	20.64	32.12
BERTSUM-ALD-MD	**35.01**	**21.65**	**33.36**
BERTSUM-ALD-ES	34.95	21.46	33.04

Tabela 5.1: Eficácia dos modelos no conjunto de dados de NCs

dade mostram que todas as abordagens geram resumos maiores que os resumos abstrativos humanos, mas com comprimento semelhante entre eles.

Resultados no conjunto de dados da WikiHow. Na Figura 5.12, são apresentados os resultados das abordagens propostas comparadas com os modelos *baselines*, em termos ROUGE-1. Percebe-se que as abordagens tem resultados consideravelmente superiores aos modelos Lead-3 e TextRank, mostrando a eficácia das mesmas. Porém, os resultados do modelo BERTSUM original foram similares aos das abordagens propostas. Isso se deve ao fato de que os textos do conjunto de dados da Wikihow possuem comprimentos, em média, menores que 512 tokens, ou seja, não necessitando na maioria dos documentos da abordagem

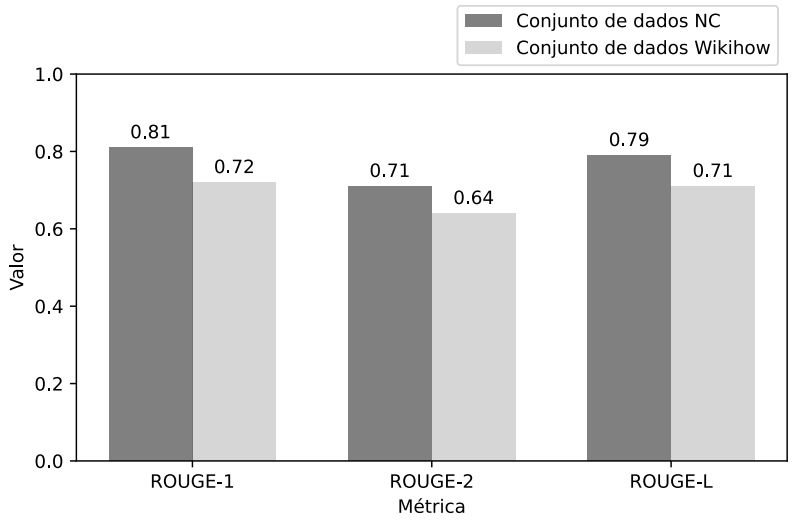


Figura 5.8: Avaliação da eficácia da abordagem BERTSUM-ALD-ES nos conjuntos de dados

Modelo	**ROUGE-1**	**ROUGE-2**	**ROUGE-L**
ORACLE	100	100	100
LEAD3	52.06	38.48	48.37
TEXTRANK	57.30	43.29	54.65
BERTSUM	65.67	51.58	65.41
BERTSUM-ALD	68.61	59.73	66.93
BERTSUM-ALD-MD	69.74	61.34	68.23
BERTSUM-ALD-ES	**69.92**	**62.47**	**68.72**

Tabela 5.2: Eficácia dos modelos no conjunto de dados da Wikihow

multi-entrada.

Na Tabela 5.2, são apresentados os resultados obtidos pelas abordagens no conjunto de dados da Wikihow. Observe que, neste caso, a abordagem BERTSUM-ALD-ES alcança os melhores resultados para todas as métricas ROUGE. O primeiro segmento apresenta as pontuações ORACLE que são as melhores pontuações para um modelo de sumarização extrativa. Vale ressaltar que as pontuações do ORACLE são perfeitas, pois os resumos de referência é composto por sentenças que fazem parte dos documentos. Essa abordagem foi aplicada devido ao fato de os resumos de referência abstrativos apresentarem baixas pontuações na métrica ROUGE.

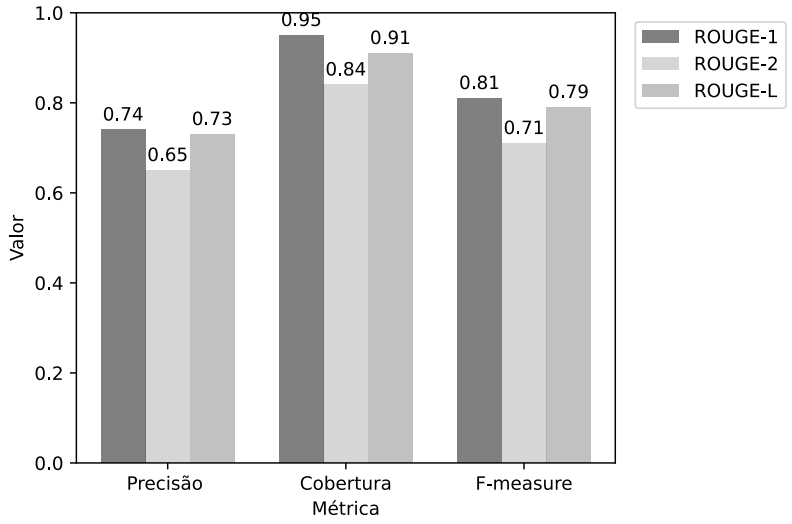


Figura 5.9: Avaliação da precisão, cobertura e $F_{measure}$ da abordagem BERTSUM-AL-ES no conjunto de dados de NCs

Como pode ser observado, no segundo segmento, dentre os modelos *baselines*, o modelo BERTSUM alcançou resultados superiores em relação aos demais. A baixa eficácia do Lead-3 sugere que não é eficaz utilizar diretamente as três primeiras sentenças dos documentos para geração dos resumos. Percebe-se que, embora o BERTSUM alcance eficácia competitiva neste conjunto de dados, ele ainda possui pontuações inferiores quando comparado com as abordagens propostas neste trabalho. Além disso, as abordagens propostas superam em pelo menos 5% os resultados dos modelos *baselines*, indicando que lidar com documentos de comprimento arbitrário pode realmente ajudar um modelo de resumo a aprender melhores representações de documentos. Por fim, a abordagem BERTSUM-ALD-ES obteve os melhores resultados para todas as métricas ROUGE, devido ao fato de que a abordagem *ensemble* normalmente aprende diferentes estilos de resumo, sem depender de técnicas de agrupamento.

O teste de *Friedman*, aplicado no conjunto das predições de cada modelo avaliado na Tabela 5.1, apresentou $estatstica = 14,159$ e $p - value = 0,013$, aceitando, assim, a hipótese alternativa de que há diferença nas distribuições das predições de cada modelo. Da mesma forma, no experimento da Tabela 5.2, o mesmo teste foi aplicado e apresentou $estatstica = 11,322$, e $p - value = 0,023$, indicando conclusão similar, a aceitação da hipótese alternativa de que há diferença nas distribuições das predições de cada modelo.

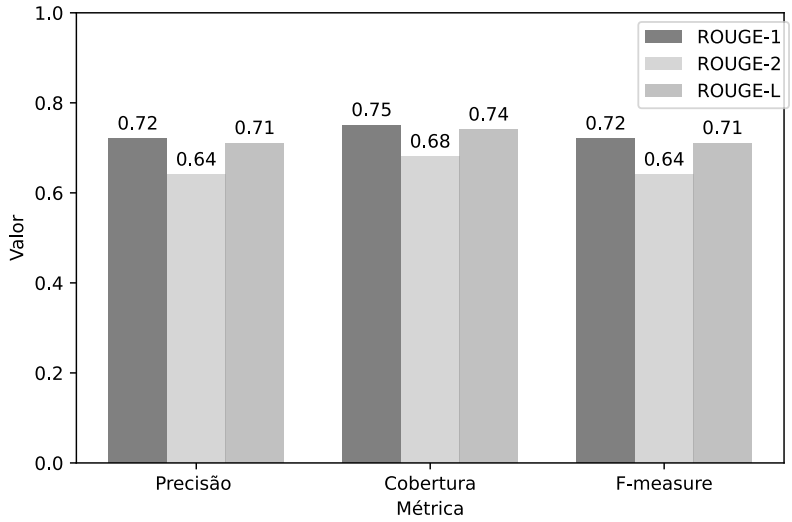


Figura 5.10: Avaliação da precisão, cobertura e $F_{measure}$ da abordagem BERTSUM-ALD-ES no conjunto de dados da Wikihow

Em resumo, e em relação à QP4, as abordagens propostas superaram os modelos *baselines* possivelmente devido aos seguintes motivos: os *baselines* são modelos mais simples, as abordagens propostas apresentam considerável generalização devido as suas características intrínsecas e, por fim, as abordagens propostas são capazes de lidar lidar de modo satisfatório, em termos de eficácia, com documentos de comprimento arbitrário dos conjuntos de dados avaliados. Além disso, a utilização das abordagens multi-domínio e *ensemble* alavancaram ainda mais os resultados nas métricas ROUGE.

5.8.2 Avaliação da Eficiência

Para mensurar a eficiência (QP5), as abordagens propostas foram avaliadas considerando o tempo de predição das sentenças mais importantes dos documentos. Na Figura 5.14, são exibidos os tempos de predição das abordagens propostas, comparados com o tempo de predição dos modelos base e estado-da-arte (BERTSUM). O eixo horizontal representa o comprimento do documento em número de caracteres e o eixo vertical representa o tempo de execução da predição em segundos. Na Figura 5.14, as curvas de densidade mostram que todas as abordagens propostas têm tempos de execução semelhantes; quanto maior o documento, maior o tempo de execução. Por outro lado, o BERTSUM apresentou tempo de execução constante independente do comprimento do documento; isso se deve ao fato de o

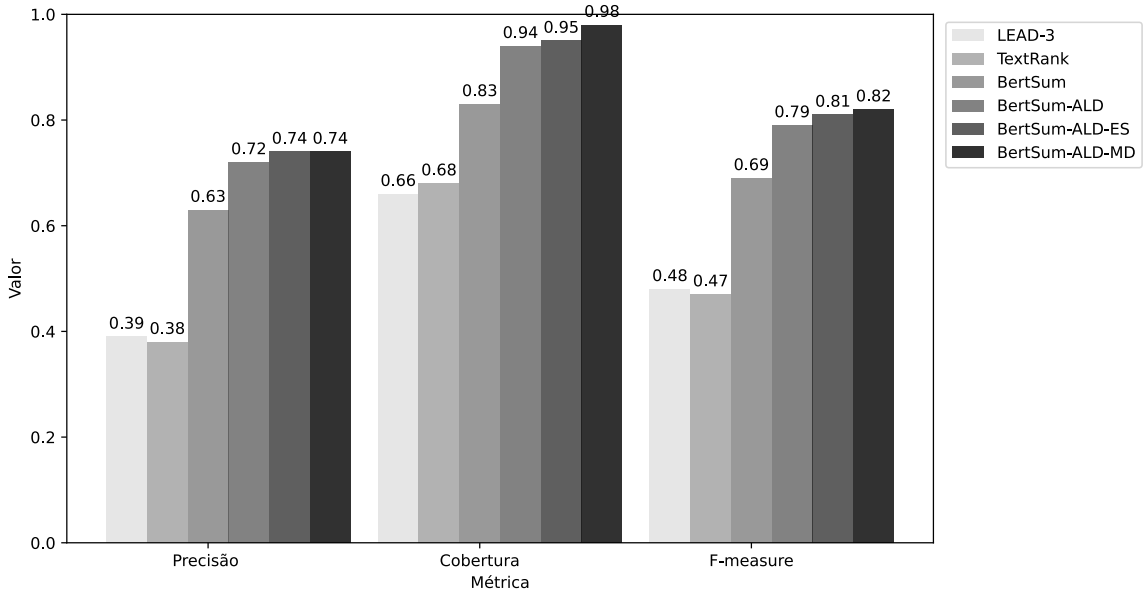


Figura 5.11: Comparação da precisão, cobertura e $F_{measure}$, em termos de ROUGE-1, das abordagens propostas com os modelos *baselines*, no conjunto de dados da Wikihow

BERTSUM processar apenas os primeiros 512 tokens do documento. Vale ressaltar que o tempo de execução das abordagens depende da paralelização dos subdocumentos a serem processados e do ambiente de execução (CPU ou GPU). Nesse experimento, não houve paralelização dos subdocumentos e a execução foi realizada sem GPU, porém em um ambiente real é necessário a paralelização do processamento e o uso de GPU para otimização do tempo de predição dos resumos.

5.9 Discussão dos Resultados

A seguir, é apresentado um resumo das conclusões obtidas em cada questão de pesquisa que guiou os experimentos.

- **QP1:** Ao avaliar a abordagem proposta de multi-entrada nos dois conjuntos de dados, os resultados indicaram que o mesmo pode ser usado para predizer as sentenças mais importantes contidas em documentos de comprimento arbitrário. Os experimentos no conjunto de dados das NCs mostraram $F_{measure} \approx 80\%$ para as métricas ROUGE-1 e ROUGE-L, e $F_{measure} \approx 70\%$ para métrica ROUGE-2. Os resultados obtidos no

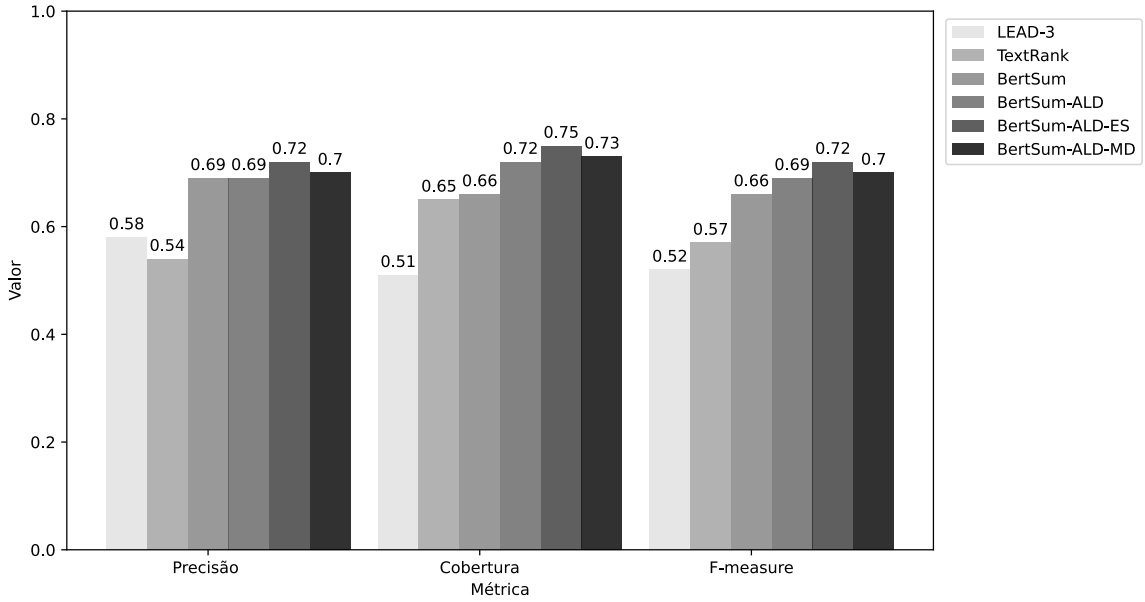


Figura 5.12: Comparação da precisão, cobertura e $F_{measure}$, em termos de ROUGE-1, das abordagens propostas com os modelos *baselines*, no conjunto de dados de NCs

conjunto de dados da Wikihow apresentaram perda de quase 10% na eficácia, onde o teste de *Mann-Whitney* indicou diferença significativa nos resultados obtidos nos dois conjuntos de dados. Essas perdas de eficácia no conjunto de dados da WIkihow podem estar relacionadas a grande quantidade de documentos na base de dados e à limitação de recursos computacionais (e.g. CPU, memória RAM) para um treinamento mais robusto dos modelos. Além disso, a maior parte dos documentos da Wikihow apresentaram resumos de referência com baixa qualidade, dificultando o trabalho do modelo em encontrar padrões que indiquem as sentenças mais importantes dos documentos.

- **QP2:** A abordagem multi-domínio proposta produziu eficácia superior a todos os modelos *baselines* e ao modelo multi-entrada. Além disso, produziu os melhores resultados no conjunto de dados de NCs, indicando que essa abordagem funciona para casos em que os documentos são de diferentes subdomínios. O teste de *Mann-Whitney*, aplicado a cada cenário, indicou diferença entre as distribuições das predições. Dessa forma, conclui-se que a utilização da abordagem multi-domínio apresenta resultados superiores aos *baselines*. Por outro lado, a abordagem multi-domínio não produziu os melhores resultados no conjunto de dados da WikiHow, possivelmente pela quali-

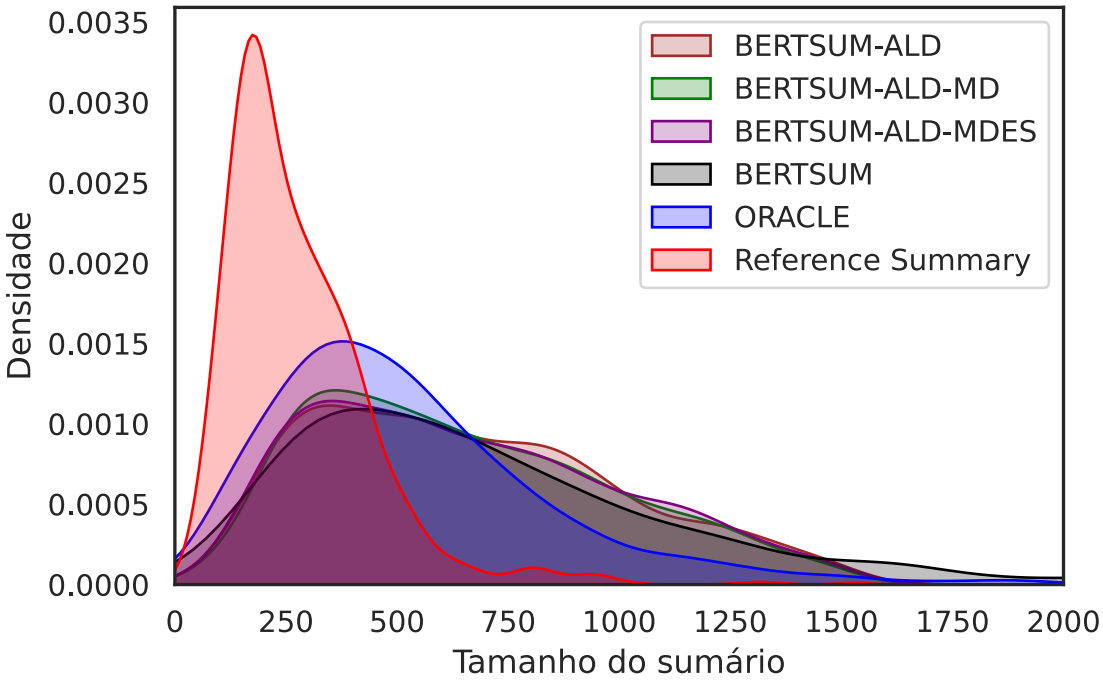


Figura 5.13: Densidade dos comprimentos dos sumários gerados pelos modelos no conjunto de dados de NCs

dade dos agrupamentos gerados que não conseguiram reunir documentos de mesma categoria no mesmo grupo.

- **QP3:** A abordagem de *ensemble* proposta produziu eficácia superior aos *baselines* e à abordagem multi-entrada, obtendo resultados inferiores apenas ao modelo multi-domínio no conjunto de dados de NCS, devido à questão da quantidade de subdomínios inerentes aos conjuntos de dados. Por outro lado, essa abordagem produziu eficácia superior as outras abordagens no conjunto de dados da Wikihow, mostrando maior adaptabilidade a diferentes conjuntos de dados. A superioridade na eficácia dessa abordagem indica uma melhor capacidade de generalização, por não necessitar de uma abordagem não supervisionada, como é o caso da abordagem multi-domínio que depende da qualidade dos agrupamentos gerados. Por fim, o teste de *Mann-Whitney*, aplicado a cada cenário, indicou diferença entre as distribuições das predições, indicando que a utilização de *ensemble* apresenta resultado superior aos *baselines* e similar ao modelo multi-domínios nos dois conjuntos de dados.

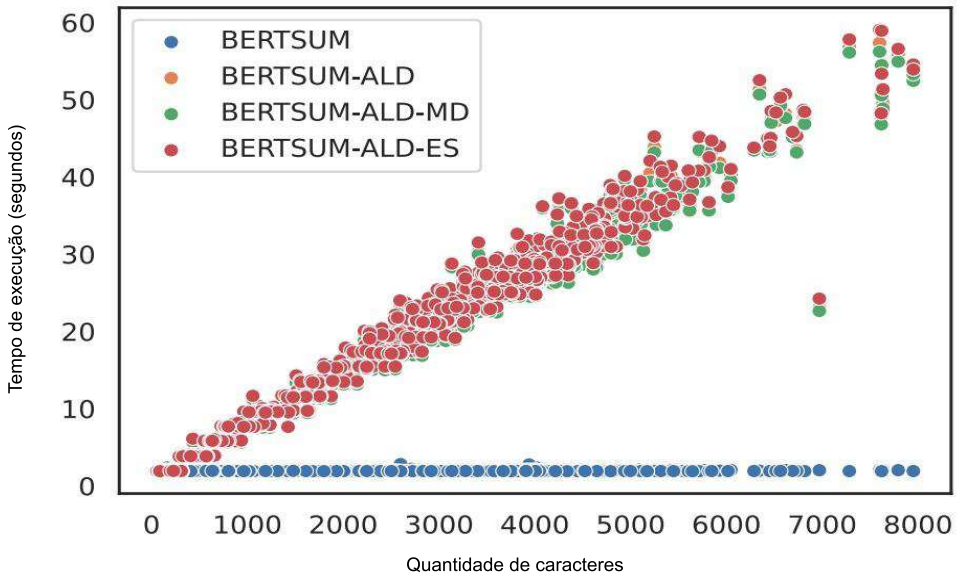


Figura 5.14: Tempo de execução, por abordagem, para a predição das sentenças mais importantes dos documentos.

- **QP4:** Com os experimentos conduzidos, observou-se que as abordagens propostas produziram resultados superiores aos *baselines* e ao modelo estado-da-arte em sumarização extrativa (BERTSUM), nos dois conjuntos de dados. Dessa forma, conclui-se que a utilização da multi-entrada, multi-domínio e *ensemble* presentes nas abordagens propostas alavancaram os resultados, mostrando a eficácia dessas técnicas.

- **QP5:** Segundo os resultados de eficiência, o tempo de execução das predições das abordagens foram similares. Porém, quando comparado ao modelo BERTSUM, o tempo de execução se mostrou maior à medida que o comprimento do documento aumenta. Isso se deve ao fato de o BERTSUM processar apenas os primeiros 512 tokens dos documentos e pelo fato da não aplicação de paralelização nas abordagens propostas. Dependendo do ambiente em que é executada a predição (CPU ou GPU), as abordagens podem ter predições mais eficientes.

5.10 Ameaças à Validade

O conjunto de dados das NCs utilizado nos experimentos contém apenas uma pequena amostra de todos as NCs presentes na Polícia Federal, devido ao fato das NCs precisarem ser extraídas manualmente da base de dados. Essa limitação impacta nos resultados das abordagens, considerando que uma quantidade reduzida de documentos inviabiliza o poder de generalização das abordagens. As abordagens tendem a recuperar as sentenças do cabeçalho devido ao fato de que o texto do cabeçalho geralmente contém palavras sobrepostas ao resumo de referência. No entanto, o cabeçalho não contém informações importantes. Isso deve ser devido à limitação na métrica ROUGE para gerar o conjunto de dados extrativo. As abordagens de extração ao nível de sentença (classificar cada sentença separadamente) não levam em consideração a qualidade do resumo como um todo. Esta limitação foi apontada em [72]. Por fim, embora os modelos propostos possam lidar com documentos de comprimento arbitrário, o contexto de cada subdocumento não é levado em consideração para prever o rótulo de uma determinada sentença, ou seja, o mecanismo de atenção das abordagens é limitado a cada subdocumento.

Os experimentos relacionados ao tempo de execução apresentam valores aproximados, visto que tais valores dependem da capacidade de processamento da máquina e da concorrência de processos por recursos. Nesta pesquisa, somente cada abordagem avaliada estava sendo executado por máquina, as abordagens foram executadas 5 vezes e foi realizada a média de tempo de execução para sumarização de cada documento.

5.11 Considerações Finais

Neste capítulo, foram apresentadas as avaliações realizadas nas abordagens propostas. Para realizar os experimentos, cinco questões de pesquisa foram definidas e motivaram a execução da avaliação experimental, a qual foi dividida em dois cenários de avaliação: eficácia e eficiência. Além disso, foram apresentados os conjuntos de dados utilizados, uma discussão dos resultados e, por fim, algumas ameaças à validade dos resultados apresentados.

No capítulo a seguir, são apresentadas as conclusões gerais da pesquisa e as perspectivas de trabalhos futuros.

Capítulo 6

Conclusões e Trabalhos Futuros

Neste capítulo, serão apresentadas as conclusões gerais deste trabalho na Seção 6.1 e as perspectivas para trabalhos futuros na Seção 6.2.

6.1 Conclusões

Este trabalho propôs diferentes abordagens para sumarização extrativa de documentos textuais. As abordagens são baseadas no modelo BERTSUM, aplicando uma abordagem multi-entrada para lidar com documentos de tamanho arbitrário. As abordagens propostas são indicadas para aplicação em conjunto de dados com documentos de tamanho arbitrário e de domínios diferentes.

O modelo BERTSUM tem sido o modelo estado-da-arte para sumarização extrativa de texto. No entanto, o BERTSUM não pode ser aplicado a tarefas de sumarização de texto longo porque não aceita como entrada, textos maiores do que o comprimento máximo de entrada do modelo BERT. Embora o comprimento máximo seja predefinido durante a etapa de pré-treinamento do modelo BERT, para expandir o comprimento máximo, o modelo precisa ser re-treinado do zero, o que geralmente requer muitos recursos de processamento e memória. Nesta pesquisa, foi explorado como usar o modelo BERT para sumarização extrativa. Foram propostas diferentes abordagens, que aproveitam o modelo BERTSUM e podem lidar com documentos com comprimento arbitrário. Além disso, foi proposto uma abordagem que utiliza de uma técnica de agrupamento automático para lidar com documento de subdomínios diferentes. Foi realizados experimentos para demonstrar como as abordagens propostas

poderiam se adequar melhor às características dos conjuntos de dados.

Considerando os experimentos realizados neste trabalho nos dois conjuntos de dados, os resultados indicaram algumas características das abordagens propostas:

- A adaptabilidade das abordagens para diferentes conjuntos de dados;
- As abordagens propostas alcançam resultados superiores do que os modelos *baselines* (TextRank, Lead-3, BERTSUM);
- Quanto mais dados, melhor a eficácia das abordagens propostas.

A abordagem BERTSUM-ALD-MD alcançou a melhor eficácia no conjunto de dados de NCs, devido ao fato de que, além da capacidade de lidar com documentos com comprimento arbitrário, a abordagem pode lidar com documentos de vários domínios. Por outro lado, no conjunto de dados da Wikihow, a abordagem BERTSUM-ALD-ES superou a abordagem BERTSUM-ALD-MD obtendo os melhores resultados. Isso deve ser devido ao fato de que esse conjunto de dados tem resumos de referência com baixa qualidade, que não agregam padrões significativos sobre diferentes domínios de texto. Além disso, foi mostrada a eficácia das abordagens propostas na tarefa de sumarização de documentos textuais realizando experimentos nos conjuntos de dados de NCs e Wikihow. Os resultados experimentais provaram que as abordagens são eficazes em tarefas de PLN com texto muito longo como entrada. Vale ressaltar que as abordagens podem ser facilmente adaptadas a vários domínios textuais. Por fim, foi analisado nos experimentos que algumas características favorecem mais uma abordagem do que as outras. Por exemplo, documentos maiores tendem a favorecer os modelos que utilizam a abordagem multi-entrada; documentos com melhor diferenciação de subdomínios tendem a favorecer modelos que utilizam a abordagem multi-domínio; documentos com menor diferenciação de subdomínios tendem a favorecer modelos que utilizam a abordagem *ensemble* e documentos com resumos de referência de baixa qualidade tendem a favorecer os modelos que utilizam a abordagem multi-entrada.

Em relação ao *tuning* das abordagens propostas, apenas foi possível avaliar os modelos com base na busca manual. Isto porque não foi possível avaliar a técnica *grid search* com a base de dados completa devido ao tempo de treinamento das abordagens: algumas semanas para *tuning* das abordagens, por exemplo; que apresentaria, portanto, os valores de

parâmetros ideais para serem conduzidos com a quantidade de dados fornecida. Dois parâmetros que tiveram grande influência nos resultados foram a taxa de aprendizagem (*learning rate*) e a taxa de decaimento (*dropout*). Foi visto que taxas de aprendizagem maiores que 0,0001 resultavam em modelos que sofriam de sub-ajuste (*underfitting*). Além disso, taxas de decaimento evitavam o modelo sofrer sobre-ajuste (*overfitting*) e sub-ajuste.

Em resumo, essas abordagens podem ajudar diretamente os agentes da Polícia Federal, provendo resumos de documentos textuais com $F_{measure} \approx 80\%$. As abordagens propostas também obtiveram melhor desempenho quando comparadas com três *baselines* disponibilizados na literatura: TextRank, Lead-3 e BERTSUM.

6.2 Trabalhos Futuros

Nesta seção, são apresentadas as perspectivas de extensão do trabalho desenvolvido nesta dissertação, como detalhado a seguir:

- **Criação de uma aplicação para resumos automáticos de documentos textuais.** Para os agentes policiais, seria interessante dispor de uma aplicação completa para sumarização e retreinamento do modelo com o *feedback* do usuário. Essa aplicação deve fornecer ao usuário a capacidade de carregar um documento de uma NC, gerar o resumo automático do documento, fornecer a possibilidade de recorreção do resumo e geração do *feedback* para aprimoramento do modelo de sumarização;

- **Aplicação de outras técnicas de ensemble.** Nesta pesquisa, foi implementada a técnica de *ensemble Bagging* sobre o modelo BERT. Porém, outras técnicas de *ensemble* para combinar os modelos-base ainda podem ser avaliadas como, por exemplo, a técnica *boosting*. Esta técnica pode gerar um modelo combinado com menos erros, pois otimiza as vantagens e reduz as armadilhas do modelo-base;

- **Implementação da Aprendizagem Incremental nos modelos propostos.** Para atualizar continuamente o modelo proposto, ou seja, incorporar novos dados ao modelo já treinado, poderia ser aplicada uma técnica de aprendizado incremental, como ilustrado na Figura 6.1. No tempo t_0 , o modelo é treinado com os dados iniciais do conjunto de dados e produz a função estimada para geração dos resumos. Uma vez treinado, o

modelo é utilizado para geração dos resumos de novos documentos; portanto, os documentos recebidos no tempo t_1 podem ser utilizados para atualizar o modelo no tempo t_2 quando houver a correção dos resumos gerados por parte de seres humanos, e assim por diante. Para isso, o modelo se adapta gradualmente, ou seja, h_{i+1} é construído com base em h_i e nas p instâncias de dados (documentos e resumos) recebidas no intervalo de tempo, sem a necessidade de um retreinamento completo e preservando o conhecimento adquirido anteriormente;

- **Retreinamento das abordagens e melhoria da eficiência.** Treinamento das abordagens com mais dados e aplicação do processamento paralelo dos subdocumentos, através do uso dos núcleos da CPU e da criação de *threads*, para melhorar ainda mais a eficácia e eficiência das abordagens propostas;

- **Avaliação dos modelos em outros conjuntos de dados.** Estudo dos resultados dos modelos em outros conjuntos de dados como, por exemplo, o conjunto de dados cnn/-dailymail [1]. O conjunto de dados cnn/dailymail é um conjunto de dados com textos em inglês contendo pouco mais de 300 mil artigos de notícias, escritos por jornalistas da CNN e do *Daily Mail*. Esse conjunto de dados pode ser utilizado para comparar o resultado das abordagens propostas com modelos estado-da-arte que foram avaliados na mesma base de dados.

[1] https://huggingface.co/datasets/cnn$_dailymail$

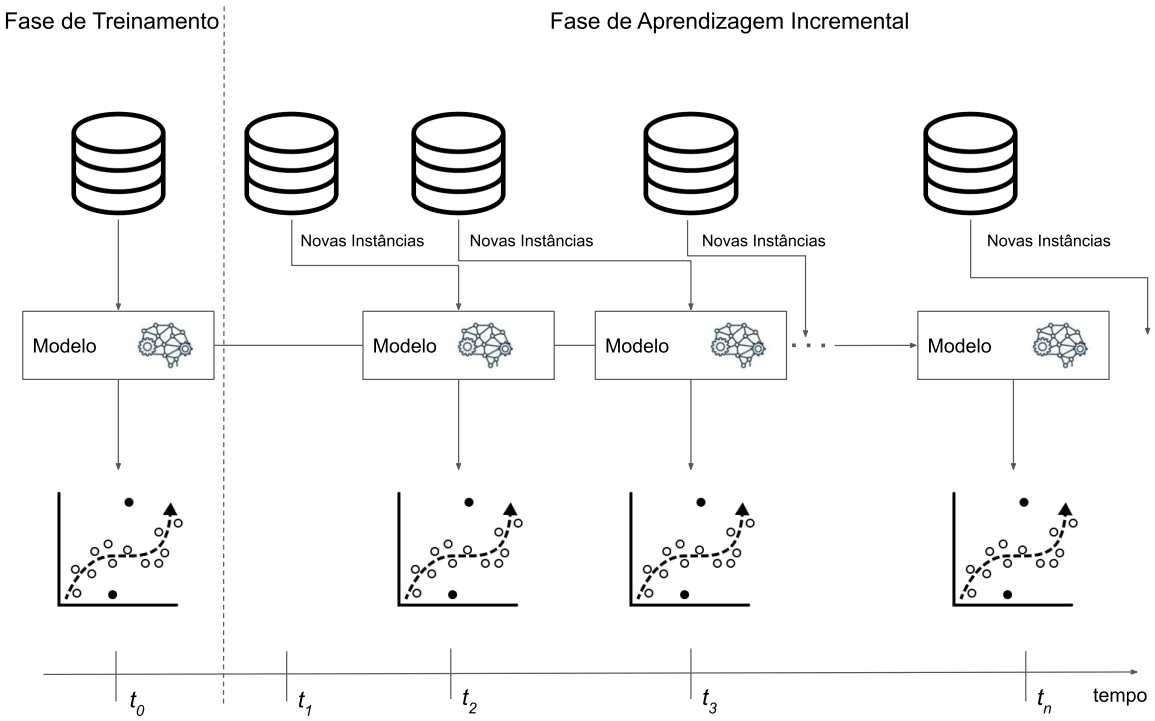


Figura 6.1: Fluxo de execução da aprendizagem incremental.

Bibliografia

[1] Mohiuddin Ahmed, Raihan Seraj, and Syed Mohammed Shamsul Islam. The k-means algorithm: A comprehensive survey and performance evaluation. *Electronics*, 9(8), 2020.

[2] Mehdi Allahyari, Seyedamin Pouriyeh, Mehdi Assefi, Saeid Safaei, Elizabeth D. Trippe, Juan B. Gutierrez, and Krys Kochut. Text summarization techniques: A brief survey. *arXiv preprint arXiv:1707.02268*, 2017.

[3] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[4] Eden Belouadah, Adrian Popescu, and Ioannis Kanellos. A comprehensive study of class incremental learning algorithms for visual tasks. *Neural Networks*, 135:38–54, 2021.

[5] Iz Beltagy, Matthew E. Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020.

[6] Abla Chouni Benabdellah, Asmaa Benghabrit, and Imane Bouhaddou. A survey of clustering algorithms for an industrial context. *Procedia Computer Science*, 148:291–302, 2019. THE SECOND INTERNATIONAL CONFERENCE ON INTELLIGENT COMPUTING IN DATA SCIENCES, ICDS2018.

[7] James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *J. Mach. Learn. Res.*, 13(null):281–305, feb 2012.

[8] Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python: analyzing text with the natural language toolkit*. "O'Reilly Media, Inc.", 2009.

[9] Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake VanderPlas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122, 2013.

[10] Peter Bühlmann. Bagging, boosting and ensemble methods. *Handbook of Computational Statistics*, 01 2012.

[11] Asli Celikyilmaz, Antoine Bosselut, Xiaodong He, and Yejin Choi. Deep communicating agents for abstractive summarization. 2018.

[12] Munehs Chandra, Vikrant Gupta, and Santosh Paul. A statistical approach for automatic text summarization by extraction. pages 268 – 271, 07 2011.

[13] Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. A discourse-aware attention model for abstractive summarization of long documents. *arXiv preprint arXiv:1804.05685*, 2018.

[14] Foreman D. I. Corder, G. W. *Nonparametric statistics: A step-by-step approach.* John Wiley & Sons, 2014.

[15] Laerth Bruno de Brito Gomes and H. Oliveira. A multi-document summarization system for news articles in portuguese using integer linear programming. 2019.

[16] Mostafa Dehghani, Stephan Gouws, Oriol Vinyals, Jakob Uszkoreit, and Łukasz Kaiser. Universal transformers. *arXiv preprint arXiv:1807.03819*, 2019.

[17] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2019.

[18] Elena Filatova and Vasileios Hatzivassiloglou. Event-based extractive summarization. In *Text Summarization Branches Out*, pages 104–111, Barcelona, Spain, July 2004. Association for Computational Linguistics.

[19] Milton Friedman. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the American Statistical Association*, 32(200):675–701, 1937.

[20] Andrea Galassi, Marco Lippi, and Paolo Torroni. Attention in natural language processing. *IEEE Transactions on Neural Networks and Learning Systems*, 32(10):4291–4308, 02 2019.

[21] Vishal Gupta and Gurpreet Lehal. A survey of text summarization extractive techniques. *Journal of Emerging Technologies in Web Intelligence*, 2, 08 2010.

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778. arXiv, 2016.

[23] Anish Jadhav, Rajat Jain, Steve Fernandes, and Sana Shaikh. Text summarization using neural networks. In *2019 International Conference on Advances in Computing, Communication and Control (ICAC3)*, pages 1–6, 2019.

[24] Karen Spärck Jones. Automatic summarising: The state of the art. *Inf. Process. Manag.*, 43:1449–1481, 2007.

[25] John D. Kelleher, Brian MacNamee, and Aoife D'Arcy. *Fundamentals of Machine Learning for Predictive Data Analytics: Algorithms, Worked Examples, and Case Studies*. MIT Press, Cambridge, MA, 2015.

[26] Memoona Khanam, Tahira Mahboob, Warda Imtiaz, Humaraia Ghafoor, and Rabeea Sehar. A survey on unsupervised machine learning algorithms for automation, classification and maintenance. *International Journal of Computer Applications*, 119:34–39, 06 2015.

[27] Farzad Kiani and Oguzhan Tas. A survey automatic text summarization. volume 5, pages 205–213, 06 2017.

[28] Vijay Kotu and Bala Deshpande. *Predictive Analytics and Data Mining: Concepts and Practice with RapidMiner*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1st edition, 2014.

[29] Mahnaz Koupaee and William Yang Wang. Wikihow: A large scale text summarization dataset. *arXiv preprint arXiv:1810.09305*, 2018.

[30] Dimitrios Kouzis-Loukas. *Learning Scrapy*. Packt Publishing Ltd, 2016.

[31] Julian Kupiec, Jan Pedersen, and Francine Chen. A trainable document summarizer. In *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '95, page 68–73, New York, NY, USA, 1995. Association for Computing Machinery.

[32] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. page 10, 01 2004.

[33] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. page 10, 01 2004.

[34] Peter J. Liu, Mohammad Saleh, Etienne Pot, Ben Goodrich, Ryan Sepassi, Lukasz Kaiser, and Noam Shazeer. Generating wikipedia by summarizing long sequences. *arXiv preprint arXiv:1801.10198*, 2018.

[35] Yang Liu. Fine-tune bert for extractive summarization. *arXiv preprint arXiv:1903.10318*, 2019.

[36] Viktor Losing, Barbara Hammer, and Heiko Wersing. Incremental on-line learning: A review and comparison of state of the art algorithms. *Neurocomputing*, 275:1261–1274, 2018.

[37] Hans Peter Luhn. The automatic creation of literature abstracts. *IBM J. Res. Dev.*, 2:159–165, 1958.

[38] Antoine Ly, Benno Uthayasooriyar, and Tingting Wang. A survey on natural language processing (nlp) and applications in insurance. 2020.

[39] H. B. Mann and D. R. Whitney. On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other. *The Annals of Mathematical Statistics*, 18(1):50 – 60, 1947.

[40] Rada Mihalcea and Paul Tarau. TextRank: Bringing order into text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 404–411, Barcelona, Spain, July 2004. Association for Computational Linguistics.

[41] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[42] Derek Miller. Leveraging bert for extractive text summarization on lectures. *arXiv preprint arXiv:1906.04165*, 06 2019.

[43] Derek Miller. Leveraging bert for extractive text summarization on lectures. *arXiv preprint arXiv:1906.04165*, 2019.

[44] Milad Moradi, Georg Dorffner, and Matthias Samwald. Deep contextualized embeddings for quantifying the informative content in biomedical text summarization. *Computer Methods and Programs in Biomedicine*, 184:105117, 2020.

[45] Rafael Francisco Marcondes de MORAES, Luiz Fernando Zambrana ORTIZ, Manoel Francisco de Barros da Motta Peixoto GIORDANI, and Rafael Francisco Marcondes de MORAES. Inquérito policial eletrônico: tecnologia, garantismo e eficiência na investigação criminal. *GIORDANI, Manoel Francisco de Barros da Motta Peixoto; MORAES, Rafael Francisco Marcondes de (Coord.). Estudos contemporâneos de polícia judiciária. São Paulo: Editora LTr*, pages 83–96, 2018.

[46] N. Moratanch and Chitrakala Gopalan. A survey on abstractive text summarization. pages 1–7, 03 2016.

[47] N. Moratanch and Chitrakala Gopalan. A survey on extractive text summarization. pages 1–6, 01 2017.

[48] Mahin Naderifar, Hamideh Goli, and Fereshteh Ghaljaei. Snowball sampling: A purposeful method of sampling in qualitative research. *Strides in Development of Medical Education*, In Press, 09 2017.

[49] Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *Thirty-first AAAI conference on artificial intelligence*, 2016.

[50] Shashi Narayan, Shay B. Cohen, and Mirella Lapata. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. *arXiv preprint arXiv:1808.08745*, 2018.

[51] Ani Nenkova and Kathleen McKeown. *Automatic Summarization*, volume 5. 06 2011.

[52] Margibel Adriana de Oliveira and Lineide do Lago Salvador Mosca. As notícias de crime: uma análise retórico-argumentativa do discurso jornalístico online por antecipação ao discurso jurídico. Master's thesis, Universidade de São Paulo, 2014.

[53] David Opitz and Richard Maclin. Popular ensemble methods: An empirical study. *J. Artif. Int. Res.*, 11(1):169–198, July 1999.

[54] Télvio Orrú, Joao Rosa, and Marcio Andrade Netto. Sabio: An automatic portuguese text summarizer through artificial neural networks in a more biologically plausible model. pages 11–20, 01 2006.

[55] Daniel W. Otter, Julian R. Medina, and Jugal K. Kalita. A survey of the usages of deep learning in natural language processing. *IEEE transactions on neural networks and learning systems*, 32(2):604–624, 2020.

[56] Jason Alan Palmer. Pdftotext, 2017.

[57] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

[58] Horst Po¨ttker. News and its communicative quality: the inverted pyramid—when and why did it appear? *Journalism Studies*, 4:501–511, 11 2003.

[59] XiPeng Qiu, TianXiang Sun, YiGe Xu, YunFan Shao, Ning Dai, and XuanJing Huang. Pre-trained models for natural language processing: A survey. *Science China Technological Sciences*, 63(10):1872–1897, Sep 2020.

[60] Dragomir Radev, Hongyan Jing, Małgorzata Styś, and Daniel Tam. Centroid-based summarization of multiple documents. *Information Processing Management*, 40:919–938, 11 2004.

[61] Lucia Rino, Thiago Pardo, Carlos Silla, Celso Kaestner, and Michael Pombo. A comparison of automatic summarizers of texts in brazilian portuguese. volume 3171, pages 235–244, 09 2004.

[62] Alexandra Savelieva, Bryan Au-Yeung, and Vasanth Ramani. Abstractive summarization of spoken andwritten instructions with bert. *arXiv preprint arXiv:2008.09676*, 08 2020.

[63] Fábio Souza, Rodrigo Nogueira, and Roberto Lotufo. *BERTimbau: Pretrained BERT Models for Brazilian Portuguese*, pages 403–417. 10 2020.

[64] José Torres. Sumarização automática de artigos científicos de engenharia de software com suporte ao processo de revisão sistemática. 09 2011.

[65] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[66] Adhika Pramita Widyassari, Supriadi Rustad, Guruh Fajar Shidik, Edi Noersasongko, Abdul Syukur, Affandy Affandy, et al. Review of automatic text summarization techniques methods. *Journal of King Saud University - Computer and Information Sciences*, 2020.

[67] Ian H. Witten, Eibe Frank, and Mark A. Hall. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 3rd edition, 2011.

[68] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32, 2020.

[69] Manzil Zaheer, Guru Guruganesh, Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. Big bird: Transformers for longer sequences. *Advances in Neural Information Processing Systems*, 33:17283–17297, 2020.

[70] Ruixuan Zhang, Zhuoyu Wei, Yu Shi, and Yining Chen. {BERT}-{al}: {BERT} for arbitrarily long document understanding. 2020.

[71] Chujie Zheng, Kunpeng Zhang, Harry Jiannan Wang, Ling Fan, and Zhe Wang. Topic-guided abstractive text summarization: a joint learning approach. *arXiv preprint arXiv:2010.10323*, 2021.

[72] Ming Zhong, Pengfei Liu, Yiran Chen, Danqing Wang, Xipeng Qiu, and Xuanjing Huang. Extractive summarization as text matching. *arXiv preprint arXiv:2004.08795*, 2020.

[73] Qingyu Zhou, Nan Yang, Furu Wei, Shaohan Huang, Ming Zhou, and Tiejun Zhao. Neural document summarization by jointly learning to score and select sentences. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 654–663, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[74] Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 109(1):43–76, 2020.

Apêndice A

Parâmetros dos Modelos

Neste Apêndice, são exibidos os valores dos parâmetros utilizados nas abrodagens para realização dos experimentos.

Nas Tabelas A.1, A.2 e A.3 são mostrados os valores dos parâmetros de cada abrodagem proposta. Tais valores foram encontrados após o ajuste dos hiper-parâmetros dos modelos considerando a técnica de busca manual e empírica.

Parâmetros	Descrição	NCs	Wikihow
Número de época	Número de vezes que todo o conjunto de dados de treinamento é mostrado à rede durante o treinamento.	10	20
Função de otimização	Métodos usados para alterar os atributos de rede neural a fim de reduzir o erro.	Adam	Adam
Taxa de Aprendizagem (*learning rate*)	Controla o quanto alterar o modelo em resposta ao erro estimado cada vez que os pesos do modelo são atualizados.	0.0001	0.002
Learning rate schedule	Método que ajusta a taxa de aprendizado entre épocas ou iterações à medida que o treinamento avança.	0.99	0.99
Tamanho do Lote (*batch size*)	Número de exemplos mostrados à rede antes que os pesos sejam atualizados.	10	10
Inicialização dos pesos da rede neural	Método para inicialização dos pesos da rede neural.	normal	normal
Número de camadas internas	Refere-se as camadas da rede neural que processam os dados de entrada e enviam a camada de saída.	1	2
Número de neurônios nas camadas internas	Geralmente o número de neurônios em uma camada controla a capacidade de representação da rede.	384	768
Funções de ativação nas camadas internas	A função de ativação decide se um neurônio deve ser ativado ou não.	relu	relu
Função de ativação na camada de saída	A função de ativação decide se um neurônio deve ser ativado ou não.	sigmoid	sigmoid
Droupout	Refere-se à inativação de neurônios em uma rede neural.	0.3	0.1

Tabela A.1: Valores dos parâmetros da abordagem BERTSUM-ALD

Parâmetros	Descrição	NCs	Wikihow
Número de época	Número de vezes que todo o conjunto de dados de treinamento é mostrado à rede durante o treinamento.	10	20
Função de otimização	Métodos usados para alterar os atributos de rede neural a fim de reduzir o erro.	Adam	Adam
Taxa de Aprendizagem (*learning rate*)	Controla o quanto alterar o modelo em resposta ao erro estimado cada vez que os pesos do modelo são atualizados.	0.0001	0.002
Learning rate schedule	Método que ajusta a taxa de aprendizado entre épocas ou iterações à medida que o treinamento avança.	0.99	0.999
Tamanho do Lote (*batch size*)	Número de exemplos mostrados à rede antes que os pesos sejam atualizados.	10	10
Inicialização dos pesos da rede neural	Método para inicialização dos pesos da rede neural.	normal	normal
Número de camadas internas	Refere-se as camadas da rede neural que processam os dados de entrada e enviam a camada de saída.	1	1
Número de neurônios nas camadas internas	Geralmente o número de neurônios em uma camada controla a capacidade de representação da rede.	768	768
Funções de ativação nas camadas internas	A função de ativação decide se um neurônio deve ser ativado ou não.	relu	relu
Função de ativação na camada de saída	A função de ativação decide se um neurônio deve ser ativado ou não.	sigmoid	sigmoid
Droupout	Refere-se à inativação de neurônios em uma rede neural.	0.1	0.1
Número de camadas de sumarização	Refere-se ao número de redes neurais no topo da saída do modelo BERT.	10	15

Tabela A.2: Valores dos parâmetros da abordagem BERTSUM-ALD-ES

Parâmetros	Descrição	NCs	Wikihow
Número de época	Número de vezes que todo o conjunto de dados de treinamento é mostrado à rede durante o treinamento.	10	20
Função de otimização	Métodos usados para alterar os atributos de rede neural a fim de reduzir o erro.	Adam	Adam
Taxa de Aprendizagem (*learning rate*)	Controla o quanto alterar o modelo em resposta ao erro estimado cada vez que os pesos do modelo são atualizados.	0.0001	0.002
Learning rate schedule	Método que ajusta a taxa de aprendizado entre épocas ou iterações à medida que o treinamento avança.	0.99	0.99
Tamanho do Lote (*batch size*)	Número de exemplos mostrados à rede antes que os pesos sejam atualizados.	10	10
Inicialização dos pesos da rede neural	Método para inicialização dos pesos da rede neural	normal	normal
Número de camadas internas	Refere-se as camadas da rede neural que processam os dados de entrada e enviam a camada de saída.	2	2
Número de neurônios nas camadas internas	Geralmente o número de neurônios em uma camada controla a capacidade de representação da rede.	768	768
Funções de ativação nas camadas internas	A função de ativação decide se um neurônio deve ser ativado ou não.	relu	relu
Função de ativação na camada de saída	A função de ativação decide se um neurônio deve ser ativado ou não.	sigmoid	sigmoid
Droupout	Refere-se à inativação de neurônios em uma rede neural.	0.3	0.3
Número de camadas de sumarização	Refere-se ao número de redes neurais no topo da saída do modelo BERT.	10	10

Tabela A.3: Valores dos parâmetros da abordagem BERTSUM-ALD-MD

Apêndice B

Tempo de Execução dos Experimentos

Neste Apêndice, são exibidos os tempos de treinamento/execução de cada abordagem nos experimentos conduzidos. Vale ressaltar que, como citado na Seção 5.10 de Ameaças a Validade, os tempos de execução exibidos a seguir são valores aproximados, já que tais valores dependem da capacidade de processamento da máquina e da concorrência de processos por recursos.

Base de dados	Tempo de treinamento	Abordagem
Wikihow	4h	BERTSUM-ALD
Wikihow	4,5h	BERTSUM-ALD-MD
Wikihow	4,5h	BERTSUM-AL-ES
NCs	7h	BERTSUM-ALD
NCs	8,3h	BERTSUM-ALD-MD
NCs	8h	BERTSUM-ALD-ES

Tabela B.1: Tempo de treinamento das abordagens propostas.

Base de dados	Tempo de experimentação	Abordagem
Wikihow	45h	BERTSUM-ALD
Wikihow	55h	BERTSUM-ALD-MD
Wikihow	55h	BERTSUM-AL-ES
NCs	85h	BERTSUM-ALD
NCs	92h	BERTSUM-ALD-MD
NCs	90h	BERTSUM-ALD-ES

Tabela B.2: Tempo de experimentação das abordagens propostas.