UNIVERSIDADE FEDERAL DE SERGIPE

CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA

PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Uma Análise Exploratória e Experimental de Métodos de Resumo Automático de Texto na Saúde

Dissertação de Mestrado

João Alysson dos Santos Guimarães

São Cristóvão – Sergipe

2026

UNIVERSIDADE FEDERAL DE SERGIPE

CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA

PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

João Alysson dos Santos Guimarães

Uma Análise Exploratória e Experimental de Métodos de Resumo Automático de Texto na Saúde

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Sergipe como requisito parcial para a obtenção do título de mestre em Ciência da Computação.

Orientador(a): Prof. Dr. Methanias Colaço Júnior

São Cristóvão – Sergipe

2026

FICHA CATALOGRÁFICA ELABORADA PELA BIBLIOTECA CENTRAL
UNIVERSIDADE FEDERAL DE SERGIPE

Guimarães, João Alysson dos Santos
G963 Uma análise exploratória e experimental de métodos de resumo automático de texto na saúde / João Alysson dos Santos Guimarães ; orientador Methanias Colaço Rodrigues Júnior - São Cristóvão, 2026.
180 f.; il.

Dissertação (mestrado em Ciência da Computação) – Universidade Federal de Sergipe, 2026.

1. Processamento de linguagem natural (Computação). 2. Resumos. 3. Inteligência artificial. 4. Mineração de dados (Computação). I. Rodrigues Júnior, Methanias Colaço orient. II. Título.

CDU 004:61

UNIVERSIDADE FEDERAL DE SERGIPE
PRÓ-REITORIA DE PÓS-GRADUAÇÃO E PESQUISA
COORDENAÇÃO DE PÓS-GRADUAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Ata da Sessão Solene de Defesa da Dissertação do Curso de Mestrado em Ciência da Computação-UFS.
Candidato: JOAO ALYSSON DOS SANTOS GUIMARAES

Em 03 dias do mês de março do ano de dois mil e vinte seis, com início às 15:15hs, realizou-se na Sala de Seminários do PROCC da Universidade Federal de Sergipe, na Cidade Universitária Prof. José Aloísio de Campos, a Sessão Pública de Defesa de Dissertação de Mestrado do candidato **JOAO ALYSSON DOS SANTOS GUIMARAES** que desenvolveu o trabalho intitulado: **"Uma Análise Exploratória e Experimental de Métodos de Resumo Automático de Texto na Saúde",** sob a orientação do Prof. Dr. **Methanias Colaço Rodrigues Junior.** A Sessão foi presidida pelo Prof. Dr. **Methanias Colaço Rodrigues Junior** (PROCC/UFS), que após a apresentação da dissertação passou a palavra aos outros membros da Banca Examinadora, o Dr. **Breno Santana Santos (UFRN)** e, em seguida, Dr. **Juciano de Sousa Lacerda** (UFRN) e na sequência o Dr. **André Britto de Carvalho** (Procc) . Após as discussões, a Banca Examinadora reuniu-se e considerou o mestrando (a) **APROVADO** "*(aprovado/reprovado)*". Atendidas as exigências da Instrução Normativa 05/2019/PROCC, do Regimento Interno do PROCC (Resolução 67/2014/CONEPE), e da Resolução nº 04/2021/CONEPE que regulamentam a Apresentação e Defesa de Dissertação, e nada mais havendo a tratar, a Banca Examinadora elaborou esta Ata que será assinada pelos seus membros e pelo mestrando.

Cidade Universitária "*Prof. José Aloísio de Campos",* 03 de março de 2026.

gov.br Documento assinado digitalmente
METHANIAS COLACO RODRIGUES JUNIOR
Data: 05/03/2026 19:11:07-0300
Verifique em https://validar.iti.gov.br

Prof. Dr. Methanias Colaço Rodrigues Junior
(PROCC/UFS)
Presidente

gov.br Documento assinado digitalmente
ANDRE BRITTO DE CARVALHO
Data: 05/03/2026 10:31:27-0300
Verifique em https://validar.iti.gov.br

Prof. Dr. André Britto de Carvalho
(PROCC/UFS)
Examinador Interno

gov.br Documento assinado digitalmente
BRENO SANTANA SANTOS
Data: 04/03/2026 17:13:47-0300
Verifique em https://validar.iti.gov.br

Dr.Breno Santana Santos
(UFRN)
Examinador Externo ao programa

gov.br Documento assinado digitalmente
JUCIANO DE SOUSA LACERDA
Data: 04/03/2026 16:19:15-0300
Verifique em https://validar.iti.gov.br

Prof. Dr.Juciano de Sousa Lacerda
(UFRN)
Examinador Externo ao programa

gov.br Documento assinado digitalmente
JOAO ALYSSON DOS SANTOS GUIMARAES
Data: 04/03/2026 11:45:48-0300
Verifique em https://validar.iti.gov.br

JOAO ALYSSON DOS SANTOS GUIMARAES
Candidato

Agradecimentos

Registro meus agradecimentos, ao meu orientador, **Prof. Dr. Methanias Colaço Rodrigues Júnior**, pela condução cuidadosa, pelo constante estímulo e pelas orientações valiosas, essenciais para o desenvolvimento e a finalização deste trabalho. Sua experiência, comprometimento e disponibilidade foram determinantes para minha formação acadêmica e profissional.

Manifesto minha gratidão à **Universidade Federal de Sergipe (UFS)** e ao **Programa de Pós-graduação em Ciência da Computação (PROCC)**, pelo suporte institucional, pelas condições acadêmicas oferecidas e pela infraestrutura disponibilizada, que viabilizaram a execução desta pesquisa.

Agradeço a todos os professores que ministraram as disciplinas ao longo do curso, pelo conhecimento compartilhado, pela dedicação ao ensino e pelas contribuições fundamentais para minha formação acadêmica.

Por fim, estendo meus agradecimentos a todos que, de forma direta ou indireta, fizeram parte desta jornada e contribuíram para a realização deste trabalho. Recebam meu sincero reconhecimento.

"May your heart
be your guiding key"
(Kingdom Hearts)

Resumo

Contexto: Fraude e corrupção estão entre os principais crimes que afetam as instituições públicas, sendo o setor de saúde particularmente vulnerável em razão de sua complexidade estrutural, da coexistência de prestadores públicos e privados, do grande número de atores envolvidos, da natureza globalizada das cadeias de suprimentos, dos elevados custos financeiros e da assimetria de informações entre as partes interessadas. Esses fatores fragilizam os sistemas de saúde, resultando em desperdício de recursos, redução da resiliência em situações de emergência médica e limitação do acesso a serviços essenciais. **Objetivo:** Este trabalho tem como objetivo geral desenvolver e avaliar Métodos de Resumo Automático de Texto, voltados ao apoio de auditorias no setor de saúde. **Métodos:** Inicialmente, foi realizado um Mapeamento Sistemático da Literatura com o objetivo de investigar os estudos mais recentes sobre sumarização de textos no domínio da saúde, por meio da análise dos métodos desenvolvidos, dos desafios enfrentados, dos benefícios alcançados e das áreas de aplicação. Em seguida, dois métodos de resumo extrativo de texto foram propostos, e, somado a isso, realizamos avaliações experimentais para as tarefas de resumo abstrativo, classificação de texto e modelagem de tópicos. **Resultados:** Os achados do Mapeamento evidenciam a predominância contínua de métodos tradicionais de sumarização, como os baseados em Linguística Computacional (73,91%), observamos também que 73,91% dos métodos utilizam abordagens híbridas, bem como a necessidade de abordagens mais escaláveis. Além disso, a maioria dos estudos aborda o problema sob a perspectiva extrativa (82%), o que evidencia uma lacuna de pesquisa no que se refere a métodos abstrativos. A ampla gama de aplicações ressalta a adaptabilidade das técnicas de sumarização de textos em diferentes campos médicos e biomédicos. Foram propostos, desenvolvidos e avaliados experimentalmente dois métodos não supervisionados baseados em otimização e algoritmos meméticos para a tarefa de resumo automático genérico extrativo multi-documento de texto, buscando mitigar as limitações identificadas na literatura, como limitações de avaliação, capacidade de generalização, complexidade computacional, dependência de especialistas, resumos baseados em *query* e utilizando apenas um documento. Os resultados das análises estatísticas indicam que as abordagens propostas são melhores que 16 métodos da literatura de algoritmos bioinspirados. Ademais, as avaliações experimentais demonstraram a viabilidade e confiabilidade no uso de métodos abstrativos para resumo de texto, classificação de texto e modelagem de tópicos para o suporte na atividade de auditoria, contribuindo para a redução da sobrecarga informacional. **Conclusão:** Em síntese, os resultados obtidos a partir de análises estatísticas indicam a viabilidade do emprego de técnicas de sumarização, classificação textual e modelagem de tópicos como ferramentas de apoio à tomada de decisão, contribuindo tanto para a mitigação da sobrecarga informacional quanto para o aprimoramento da eficiência das atividades analíticas desempenhadas por auditores.

Palavras-chave: Processamento de Linguagem Natural. Resumo de Texto. Classificação de Texto.

Modelagem de Tópicos. Saúde Pública. Auditoria.

Abstract

Context: Fraud and corruption are among the main crimes that affect public institutions, with the health sector being particularly vulnerable due to its structural complexity, the coexistence of public and private providers, the large number of actors involved, the globalized nature of supply chains, the high financial costs and the asymmetry of information between interested parties. These factors weaken health systems, resulting in wasted resources, reduced resilience in medical emergencies and limited access to essential services. **Objective:** The general objective of this work is to develop and evaluate Automatic Text Summary Methods, aimed at supporting audits in the health sector. **Methods:** Initially, a Systematic Literature Mapping was carried out with the aim of investigating the most recent studies on text summarization in the health domain, through the analysis of the methods developed, the challenges faced, the benefits achieved and the areas of application. Then, two methods for extractive text summarization were proposed, and, in addition, we carried out experimental evaluations for the tasks of abstractive summarization, text classification and topic modeling. **Results:** The Mapping findings highlight the continued predominance of traditional summarization methods, such as those based on Computational Linguistics (73.91%), we also observe that 73.91% of the methods use hybrid approaches, as well as the need for more scalable approaches. Furthermore, the majority of studies approach the problem from an extractive perspective (82%), which highlights a research gap regarding abstractive methods. The wide range of applications highlights the adaptability of text summarization techniques in different medical and biomedical fields. Two unsupervised methods based on optimization and memetic algorithms were proposed, developed and experimentally evaluated for the task of automatic generic extractive multi-text document summarization, seeking to mitigate the limitations identified in the literature, such as evaluation limitations, generalization capacity, computational complexity, dependence on experts, summaries based on *query* and using only one document. The results of the statistical analyzes indicate that the proposed approaches are better than 16 methods from the bioinspired algorithm literature. Furthermore, experimental evaluations demonstrated the feasibility and reliability of using abstractive methods for text summarization, text classification and topic modeling to support audit activity, contributing to the reduction of information overload. **Conclusion:** In summary, the results obtained from statistical analyzes indicate the feasibility of using summarization, textual classification and topic modeling techniques as tools to support decision-making, contributing both to mitigating informational overload and improving the efficiency of analytical activities performed by auditors.

Keywords: Natural Language Processing. Text Summarization. Text Classification. Topic Modeling. Public Health. Public Audit.

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

ACM	Association for Computing Machinery
AD	Anderson-Darling
ANN	Artificial Neural Network
ATS	Automatic Text Summarization
AudSUS	Auditoria do Sistema Único de Saúde
BART	Bidirectional and Auto-Regressive Transformers
BERT	Bidirectional Encoder Representations from Transformers
BiLSTM	Bidirectional Long Short-Term Memory
BLEU	Bilingual Evaluation Understudy
CAPES	Coordenação de Aperfeiçoamento de Pessoal de Nível Superior
CBR	Case-Based Reasoning
CL	Computational Linguistics
CNN	Convolutional Neural Network
CNPMI	Normalized Pointwise Mutual Information
CRF	Conditional Random Fields
CV	Coherence Value
DCOMP	Departamento de Computação
DE	Differential Evolution
DL	Deep Learning
DSR	Document Summary Records
DUC	Document Understanding Conference
EC	Exclusion Criteria
EHR	Electronic Health Record
EM	Expectation Maximization

EMR	Electronic Medical Record
FinBERT	Financial Bidirectional Encoder Representations from Transformers
FN	False Negative
FP	False Positive
GPT	Generative Pre-trained Transformer
GQM	Goal Question Metric
HDP	Hierarchical Dirichlet Processes
HSSFLA	Holistic Text Summarization with the Shuffled Frog-Leaping Algorithm
IA	Inteligência Artificial
IC	Inclusion Criteria
IEEE	Institute of Electrical and Electronics Engineers
IF	Isolation Forest
ILP	Integer Linear Programming
IML	Interactive Machine Learning
IQR	Interquartile Range
KDT	Knowledge Discovery in Text
KS	Kolmogorov-Smirnov
LDA	Latent Dirichlet Allocation
LLM	Large Language Model
LSA	Latent Semantic Analysis
LSTM	Long Short-Term Memory
MAE	Mean Absolute Error
METEOR	Metric for Evaluation of Translation with Explicit Ordering
ML	Machine Learning
MLP	Multi Layer Perceptron
MRMRSFLA	Maximum Relevance with Minimum Redundancy using the Shuffled Frog-Leaping Algorithm

MSL	Mapeamento Sistemático da Literatura
mT5	Multilingual Text-to-Text Transfer Transformer
NER	Named Entity Recognition
NLG	Natural Language Generation
NLP	Natural Language Processing
NLTK	Natural Language Toolkit
NLU	Natural Language Understanding
NMF	Non-negative Matrix Factorization
NPAD	Núcleo de Processamento de Alto Desempenho
OMS	Organização Mundial da Saúde
PAACDA	Proximity-based Adamic Adar Corruption Detection Algorithm
PE	Pernambuco
PICO	Population, Intervention, Control and Outcome
PLN	Processamento de Linguagem Natural
pLSA	Probabilistic Latent Semantic Analysis
POS	Part-of-Speech
PRIMERA	Pyramid-based Masked Sentence Pre-training for Multi-document Summarization
PRISMA	Preferred Reporting Items for Systematic Reviews and Meta-Analyses
PROPOR	International Conference on Computational Processing of Portuguese
RF	Random Forest
RoBERTa	Robustly Optimized BERT Pretraining Approach
ROUGE	Recall-Oriented Understudy for Gisting Evaluation
ROUGE-L	Recall-Oriented Understudy for Gisting Evaluation – Longest Common Subsequence
RQ	Research Question
SBSI	Simpósio Brasileiro de Sistemas de Informação

SFLA	Shuffled Frog-Leaping Algorithm
SLMs	Small Language Models
SUS	Sistema Único de Saúde
SVC	Support Vector Classifier
SVM	Support Vector Machine
TF-IDF	Term Frequency–Inverse Document Frequency
TM	Text Mining
TN	True Negative
TP	True Positive
UFRN	Universidade Federal do Rio Grande do Norte
UFS	Universidade Federal de Sergipe
UK	United Kingdom
USA	United States of America

Lista de símbolos

Σ	Letra grega sigma maiúscula
α	Letra grega alfa
$\in$	Pertence
$\subset$	Está contido ou subconjunto
k	Letra grega kappa
γ	Letra grega gamma
β	Letra grega beta maiúscula

Sumário

1 Introdução

Este capítulo apresenta uma contextualização do tema de pesquisa, abordando a motivação, a problemática, as questões de pesquisa, os objetivos e as hipóteses testadas.

1.1 Contextualização

O avanço das tecnologias digitais tem transformado significativamente as dinâmicas de mercado, intensificando a competitividade entre organizações. Esse progresso tecnológico impulsiona inovações que aprimoram processos de negócios, desenvolvimento de produtos e engajamento com o cliente. Além disso, proporciona redução nos custos de transação e aumento da atividade econômica (LABAZANOVA; BOTSIEVA; PERYAKINA, 2023; MURAT; SAIDA; DZHAMILYA, 2023). Observa-se também a otimização nos processos de produção e na cadeia de suprimentos, resultando em maior agilidade e resiliência dos negócios. A integração de tecnologias digitais no desenvolvimento de produtos e serviços tem elevado a satisfação dos clientes (BERAWI et al., 2020; KOTELNIKOVA, 2022), enquanto a adoção de marketing digital tem sido crucial para as empresas recuperarem e fortalecerem seu posicionamento no mercado (SILVA; MAMEDE; SANTOS, 2022).

Nesse contexto, a inovação orientada por dados emerge como um fator estratégico para o crescimento organizacional, proporcionando vantagens competitivas significativas por meio do desenvolvimento de produtos baseados em dados e capacidades analíticas aprimoradas (BLAGOEVA; BELSOSKA, 2019; SARITAS et al., 2021).

Esse avanço tecnológico impulsionou o surgimento das redes sociais, resultando em um crescimento exponencial do volume de dados gerados nos últimos anos (SANCHEZ-GOMEZ; VEGA-RODRíGUEZ; PéREZ, 2022). Consequentemente, informações sobre oportunidades de negócios, novas regulamentações, concorrentes e clientes tornaram-se abundantes e dispersas em um número crescente de fontes, como notícias, boletins internos, relatórios de mercado e redes

sociais (WALTINGER et al., 2013). Essas fontes oferecem às empresas oportunidades valiosas para aprimorar sua competitividade, permitindo-lhes adaptar-se rapidamente às necessidades de seus consumidores ou posicionar-se estrategicamente diante de seus concorrentes. Plataformas como Twitter e Facebook, por exemplo, tornaram-se cruciais devido ao conteúdo gerado pelos usuários, que inclui opiniões e sentimentos dos consumidores (KIM et al., 2016), além de possibilitar a identificação de tendências de mercado (HAN; HAO; HUANG, 2018).

Assim, a capacidade de processar grandes volumes de dados (*Big Data*) tornou-se essencial em diversas áreas acadêmicas e setores econômicos, abrangendo disciplinas como Física e Biologia, além de setores como finanças, saúde e políticas públicas. No entanto, devido à natureza pouco estruturada e incompleta desses dados, informações importantes podem permanecer inacessíveis aos usuários. Nesse contexto, o desenvolvimento de tecnologias e ferramentas avançadas para localizar, transformar, analisar e visualizar dados é fundamental, permitindo torná-los acessíveis e úteis para a tomada de decisões de forma eficiente e precisa (SOWMYA et al., 2017).

1.2 Problema de Pesquisa

Nesse contexto, como a maior parte dos dados é desestruturada e textual, o Processamento de Linguagem Natural (PLN), usualmente chamado de *Natural Language Processing* (NLP) tem sido gradualmente aplicado na administração pública. Governos e instituições públicas têm utilizado essa tecnologia para processar grandes volumes de documentos, com o objetivo de melhorar a qualidade dos serviços públicos, aumentar a confiança dos cidadãos nas instituições, e aprimorar a eficiência e eficácia do trabalho, especialmente em áreas funcionais como saúde, educação e tomada de decisão (JIANG et al., 2023).

A maior parte dos dados gerados atualmente são desestruturados e textuais, e diante disso, o uso de NLP tem sido gradualmente aplicado na administração pública. Governos e instituições públicas têm utilizado essa tecnologia para processar grandes volumes de documentos, com o objetivo de melhorar a qualidade dos serviços públicos, aumentar a confiança dos cidadãos nas instituições, e aprimorar a eficiência e eficácia do trabalho, especialmente em áreas funcionais como saúde, educação e no processo de tomada de decisão (JIANG et al., 2023).

Muitos governos precisam analisar, em tempo real, múltiplas fontes de informação, tanto estáticas quanto dinâmicas, para monitorar câmeras públicas e privadas, comentários de cidadãos em redes sociais, transações *online* e eventos. Essa análise visa identificar padrões, correlações e estabelecer modelos preditivos que possibilitem a otimização de estratégias e a melhoria dos serviços oferecidos aos cidadãos. Outro objetivo fundamental é garantir o monitoramento e a vigilância necessários para proteger a população e mitigar o impacto de crimes (BENJELLOUN et al., 2015).

Dentre os crimes, destacam-se fraudes e corrupção, sendo o setor de saúde particularmente

vulnerável devido a diversos fatores. Entre eles, estão a complexidade dos sistemas de saúde, que combinam provedores públicos e privados; o grande número de pessoas envolvidas; a natureza globalizada da cadeia de suprimentos; os elevados gastos públicos e privados; e a assimetria de informação entre os atores, que pode impactar negativamente a tomada de decisões no setor. Essas vulnerabilidades enfraquecem os sistemas de saúde, resultando no desperdício de recursos, redução da resiliência dos países em situações de emergência médica e comprometimento da cobertura e do acesso a serviços básicos de saúde (MACKEY et al., 2018).

Um estudo da Transparência Internacional, uma organização global da sociedade civil contra a corrupção, revelou que, em 42 dos 109 países pesquisados, mais de 50% dos cidadãos acreditam que o setor de saúde em seu país é corrupto ou muito corrupto. Além disso, a Organização Mundial da Saúde (OMS) estimou que, dos US$ 7,5 trilhões gastos em saúde mundialmente em 2008, US$ 415 bilhões (7,3%) foram perdidos devido a fraudes ou corrupção no setor (MACKEY et al., 2018).

No Brasil, para o exercício de 2025, foram sancionados R$ 245,1 bilhões bilhões pelo Ministério da Saúde (Senado Federal, 2025). A do orçamento tripartite total para o SUS totalizou certa de R$ 500 a R$ 550 bilhões em 2024, já em em 2023, o gasto público total consolidado foi de R$ 454,46 bilhões (Conselho Nacional de Secretários de Saúde, 2024). Embora o SUS atenda mais de 70% da população, o gasto público representa apenas cerca de 40% do gasto total com saúde no Brasil (incluindo o setor privado) (CARVALHO, 2024).

Em contrapartida, a Controladoria-Geral da União (CGU) identificou uma distorção contábil em uma auditoria referente a 2023 de R$ 44,2 bilhões nas contas do Ministério da Saúde (Controladoria-Geral da União, 2024). Essa distorção na contabilidade abrange desde falhas de monitoramento e erros de lançamento até indícios de irregularidades graves, ou seja, recursos sem a devida comprovação ou controle de eficácia (METRÓPOLES, 2024).

No contexto operacional e financeiro, a Saúde é vulnerável à manipulação de licitações, faturamento fraudulento de seguros e adulterações na cadeia de suprimento, com a corrupção manifestando-se por meio da formação de cartéis, faturas superfaturadas e entregas de projetos e infraestrutura de qualidade inaceitável (REY-PUECH; BALABANOVA; MCKEE, 2025).

A falta de responsabilização e o desvio de recursos comprometem criticamente a capacidade dos profissionais da linha de frente de agirem de forma eficaz durante crises, como evidenciado na resposta à pandemia de COVID-19 (OBI et al., 2025).

Os impactos da corrupção vão além das perdas financeiras, abrangendo também consequências sociais, especialmente em países de baixa renda. Nessas regiões, os efeitos imediatos e de longo prazo incluem maior morbidade e mortalidade, devido às barreiras criadas pela corrupção no acesso aos serviços de saúde, afetando particularmente os grupos mais vulneráveis. A corrupção compromete a qualidade dos sistemas de saúde e distorce a alocação de investimentos no setor (MACKEY et al., 2018).

Fraudes também prejudicam a reputação e a confiança nas organizações, tornando vital a implementação de estratégias para prevenir, detectar e mitigar esses riscos. Um dos mecanismos eficazes de combate a fraudes é a utilização de ouvidorias e canais de denúncia, que desempenham um papel central nos sistemas de conformidade, permitindo o recebimento e tratamento de denúncias de fraude e corrupção (PAULA et al., 2024).

Além disso, auditorias são outra ferramenta crucial para mitigar esses crimes e seus impactos. Contudo, o grande volume de dados apresenta desafios significativos, incluindo a sobrecarga informacional, o que torna o processo de auditoria complexo e difícil de ser conduzido (AMARAL et al., 2020; PAULA et al., 2024).

O processo de auditoria é, em geral, custoso, demorado e envolve recursos humanos e materiais substanciais. Por isso, é necessário implementar soluções e técnicas que automatizam a análise de denúncias de corrupção. Esse processo geralmente se divide em duas etapas: na primeira, busca-se identificar elementos e evidências de corrupção, como fornecedores, contratos, funcionários, clientes e outras partes interessadas, avaliando a plausibilidade e consistência das denúncias e indícios de fraudes; na segunda etapa, ocorre a investigação propriamente dita (PAULA et al., 2024).

Para a construção do conhecimento necessário para o trabalho de auditoria, deve ocorrer o levantamento de informações sobre o objetivo da auditoria. Nessa etapa, são utilizadas diversas fontes, dentre elas, sites da internet (FONTES et al., 2023). Para auxiliar no processo de coleta de informações, técnicas de *webs crapping* para coleta massiva de informações de sites no contexto da saúde podem ser empregadas. Para auxiliar na análise dessa grande massa de dados, técnicas de NLP como sumarização de texto podem ser aplicadas, reduzindo o tempo e os recursos necessários para a análise e coleta de evidências de possíveis irregularidades (BENJELLOUN et al., 2015; MADUREIRA; POPOVIČ; CASTELLI, 2021).

No contexto da auditoria em saúde, a utilização eficaz de notícias como fonte de informação para direcionar investigações apresenta desafios significativos. Apesar dos avanços tecnológicos, os auditores enfrentam obstáculos consideráveis ao tentar coletar, analisar e interpretar o vasto volume de informações disponíveis em fontes de notícias. Este processo é particularmente complexo devido a:

- Volume de dados: a quantidade massiva de notícias geradas diariamente sobre o setor de saúde torna a análise manual impraticável e propensa a erros.
- Diversidade de fontes: as informações relevantes estão dispersas em uma ampla variedade de plataformas de notícias, desde grandes veículos de mídia até *blogs* especializados, dificultando uma coleta abrangente.
- Velocidade da informação: a rápida disseminação de notícias exige um processamento ágil para que as informações sejam úteis em tempo hábil para as auditorias.

- Relevância e confiabilidade: identificar notícias verdadeiramente relevantes e confiáveis em meio a um mar de informações é um desafio crítico para os auditores.
- Conexão com dados de auditoria: estabelecer ligações significativas entre as notícias e os dados específicos de auditoria requer um nível de análise contextual que os métodos tradicionais têm dificuldade em proporcionar.
- Limitações de recursos: os departamentos de auditoria frequentemente operam com recursos limitados, tornando inviável a dedicação de pessoal exclusivamente para o monitoramento e análise contínua de notícias.

Essas dificuldades resultam em um aproveitamento subótimo das notícias como recurso de inteligência para auditorias em saúde. Consequentemente, informações valiosas que poderiam direcionar investigações mais eficazes e identificar áreas de risco precocemente são frequentemente negligenciadas ou descobertas tardiamente. Há, portanto, uma necessidade premente de desenvolver métodos que possam automatizar e otimizar o processo de extração de *insights* relevantes de notícias para apoiar as atividades de auditoria no setor de saúde.

Desta forma, esta pesquisa pretende responder aos seguintes questionamentos:

- Quais métodos de resumo automático de texto são empregadas no contexto da saúde?
- É possível utilizá-las na área da saúde apoiada por notícias?
- A utilização desses métodos auxiliaria no processo de tomada de decisões dos auditores?

Para responder às questões de pesquisa, um mapeamento sistemático da literatura foi realizado, além disso, foram propostos métodos de sumarização de texto, seguido de três experimentos *in vitro* para avaliar os métodos de Resumo de Texto Automático e métodos auxiliares como Classificação de Texto e Modelagem de Tópicos da literatura no contexto de Auditoria na Saúde Pública.

1.3 Justificativa

A relevância desta dissertação excede a resolução de um desafio técnico. Sua principal justificativa se alicerça na resolução de um desafio com relevância social significativa: o suporte a auditorias na Saúde Pública. Além disso, contribuímos com a comunidade de PLN ao aumentarmos o escopo da base experimental nas tarefas de resumo, classificação e modelagem de tópicos.

Ao desenvolver e propor métodos de resumos extrativos genéricos, disponibilizamos meios concretos para resolução do problema de sobrecarga informacional nas etapas de auditoria. Somado a isso, por meio dos diversos experimentos controlados, que utilizaram métodos

experimentais robustos e avaliações estatísticas sistemáticas em larga escala, demonstramos a por meio dos resultados a viabilidade da utilização de tais métodos para auxiliar a atividade de auditoria.

1.4 Objetivo Geral

Este trabalho tem como objetivo geral desenvolver e avaliar Métodos de Resumo Automático de Texto, voltados ao apoio de auditorias no setor de saúde.

1.5 Objetivos Específicos

Para alcançar o objetivo geral, foram estabelecidos os seguintes objetivos específicos:

- Conduzir um Mapeamento Sistemático da Literatura (MSL) para identificar e analisar os métodos mais recentes de sumarização de texto no domínio da saúde;
- Desenvolver métodos extrativos de sumarização automática de texto;
- Realizar um experimento controlado *in vitro* para avaliar os algoritmos de estado da arte em resumo automático de texto abstrativos, aplicados a notícias referentes à área de saúde pública.
- Realizar experimentos controlados *in vitro* com métodos auxiliares à tarefa de Resumo Automático de Texto: Classificação de Texto e Modelagem de Tópicos.

1.6 Metodologia

A metodologia deste trabalho foi estruturada em três etapas principais. A primeira consistiu em um mapeamento sistemático da literatura (MSL), submetido ao XXII Simpósio Brasileiro de Sistemas de Informação (SBSI). Este mapeamento teve como objetivo identificar o estado da arte das pesquisas sobre métodos de resumo automático de texto aplicados no contexto da saúde, seus desafios enfrentados, benefícios alcançados e suas pareas de aplicação; o artigo está detalhadamente descrito no Capítulo 3.

Visando alcançar o objetivo principal da pesquisa, a segunda etapa consistiu na proposição de métodos de resumo de texto. A terceira etapa, consistiu na execução de um experimento controlado *in vitro*. Este experimento utilizará uma base de dados de notícias relacionadas à área da saúde. O processo de construção da base é descrito detalhadamente na Subseção 2.9.1.

1.7 Principais Contribuições

As principais contribuições desta dissertação consistem na demonstração da viabilidade de adoção de métodos de resumo extrativo e abstrativo, bem como de métodos auxiliares, tais como classificação de texto e modelagem de tópicos, para a mitigação do problema de sobrecarga informacional no contexto de auditorias em saúde, especialmente na fase analítica voltada à identificação de indícios de irregularidades, por meio de análises estatísticas e avaliações experimentais.

Para tanto, foram conduzidos estudos independentes, nos quais cada etapa foi delineada com rigor metodológico e consistência estatística, assegurando a replicabilidade dos experimentos a partir da base de dados pública disponibilizada e do código-fonte aberto. Adicionalmente, destacam-se as seguintes contribuições, materializadas em seis produções científicas:

1. **Mapeamento Sistemático da Literatura**: *A Systematic Mapping of Text Summarization Methods Applied in Health Domain*, submetido para o XXII Simpósio Brasileiro de Sistemas de Informação (SBSI).

2. **Métodos de resumo extrativo:** Proposição e avaliação de novos métodos de resumo extrativos MRMRSFLA e HSSFLA:.

    2.1. O MRMRSFLA foi introduzido no artigo *A Topic Based Generic Extractive Multi-document Text Summarization Method Using Memetic Algorithm and Combinatorial Optimization*, submetido ao periódico *Memetic Computing*;

    2.2. Enquanto que o HSSFLA foi apresentado no artigo *A Generic Extractive Multi-document Text Summarization Method Using Memetic Algorithm and Combinatorial Optimization*, **aceito** pelo XXII Simpósio Brasileiro de Sistemas de Informação (SBSI).

3. **Avaliação Experimental:** Os 3 artigos que avaliam experimentalmente SLMs, métodos de classificação e modelagem de tópicos expandem a base de conhecimento experimental:

    3.1. *Small Language Models Applied in Text Summarization Task of Health-Related News to Improve Public Health Audit: An Experimental Case Study*, **publicado** no periódico *Frontiers in Artificial Intelligence*;

    3.2. *Experimental Evaluation of Machine Learning Algorithms for Classifying Health-Related News with Indications of Irregularity*, **aceito** no XXII Simpósio Brasileiro de Sistemas de Informação (SBSI);

    3.3. *Experimental Evaluation of Topic Modeling Methods for Categorizing Irregularities in Health-related news* **aceito** na conferência 7th International Conference on Computational Processing of Portuguese (PROPOR 2026).

1.8 Organização da Dissertação

Este documento está organizado de acordo com a Instrução Normativa Nº 01/2023/PROCC, que permite que a dissertação seja uma "compilação de artigos científicos publicados ou submetidos em veículos com Qualis Restrito e ter seu conteúdo apresentado em formato alternativo". Os tópicos a seguir descrevem o conteúdo de cada um dos Capítulos:

No Capítulo 1, apresenta-se a Introdução, na qual são discutidas a contextualização do tema, a delimitação do problema de pesquisa, a definição dos objetivos geral e específicos, bem como a metodologia adotada. Nesse capítulo, são abordados os desafios enfrentados no campo da saúde, com ênfase nas atividades de auditoria no Sistema Único de Saúde (SUS). A partir desse contexto, formulam-se as perguntas de pesquisa e os objetivos do estudo, propondo-se o enfrentamento do problema por meio da aplicação de métodos de resumo automático de texto.

No Capítulo 2, é apresentada a Fundamentação Teórica, com a conceituação dos principais temas que sustentam a pesquisa. São abordados o campo de Processamento de Linguagem Natural, bem como os conceitos de Resumo Automático de Texto, Algoritmos bioinspirados, Classificação de Texto, Modelagem de Tópicos e Modelos de Linguagem. Adicionalmente, são descritas as métricas de avaliação empregadas ao longo do estudo.

No Capítulo 3, é apresentada a replicação parcial de um mapeamento sistemático da literatura submetido ao XXII Simpósio Brasileiro de Sistemas de Informação (SBSI). Nesse capítulo, discutem-se os benefícios e os desafios identificados na literatura acerca do uso de técnicas de resumo automático de texto no contexto da saúde. Os achados e as lacunas de pesquisa identificadas nesse mapeamento orientaram o desenvolvimento dos capítulos subsequentes.

No Capítulo 4, são discutidos os trabalhos relacionados à algoritmos bioinspirados com foco em resumo automático de texto, apresentando estudos e pesquisas relevantes que fundamentam os métodos de resumo extrativo propostos nos capítulos posteriores.

Os Capítulos 5 e 6 replicam parcialmente os artigos dos métodos propostos de resumo de texto genérico extrativo e multidocumento, respectivamente *A Topic Based Generic Extractive Multi-document Text Summarization Method Using Memetic Algorithm and Combinatorial Optimization*, submetido ao periódico *Memetic Computing*, e *A Generic Extractive Multi-document Text Summarization Method Using Memetic Algorithm and Combinatorial Optimization*, submetido ao XXII Simpósio Brasileiro de Sistemas de Informação (SBSI).

Com base nos achados do mapeamento, nos Capítulos 7, 8 e 9, são reproduzidos parcialmente os experimentos controlados conduzidos para as tarefas de Processamento de Linguagem Natural de Resumo de Texto, Classificação de Texto e Modelagem de Tópicos. Esses estudos foram submetidos, respectivamente, ao periódico *Frontiers in Artificial Intelligence* e às conferências XXII Simpósio Brasileiro de Sistemas de Informação (SBSI) e 17th International Conference on Computational Processing of Portuguese (PROPOR 2026).

No Capítulo 10 apresentamos uma discussão deste trabalho, descrevendo as lacunas da literatura, conectando os resultados de cada pesquisa individual apresentada e justificando as decisões tomadas.

Por fim, no Capítulo 11, são apresentadas as conclusões do trabalho, destacando-se as principais contribuições, limitações e direções para pesquisas futuras. Esse capítulo sintetiza os principais resultados e discute as implicações práticas e teóricas da aplicação de técnicas de resumo automático, classificação de texto e modelagem de tópicos como suporte às atividades de auditoria no setor da saúde.

2

Fundamentação Teórica

Este capítulo apresenta parte do arcabouço teórico essencial para fundamentar a presente pesquisa. Serão abordados e definidos os conceitos centrais de Processamento de Linguagem Natural (PLN), Resumo de Texto Automático, Classificação de Texto e Modelagem de Tópicos, estabelecendo assim a base conceitual para o desenvolvimento subsequente do trabalho.

2.1 Processamento de Linguagem Natural

O campo de pesquisa do Processamento de Linguagem Natural (PLN) tem como objetivo investigar e propor métodos e sistemas de processamento computacional da linguagem humana. Na área da Ciência da Computação, o PLN está ligado à área de Inteligência Artificial (IA), assim como à Linguística Computacional. Esse campo busca soluções para problemas computacionais, sejam tarefas, sistemas, aplicações ou programas, que requerem o tratamento computacional de uma língua como o português ou inglês, seja escrita (texto) ou falada (CASELI; NUNES, 2024). E ele é dividido em duas subáreas:

- *Natural Language Understanding* (NLU) ou Interpretação (ou Compreensão) de Linguagem Natural, que foca na análise e interpretação da língua.
- *Natural Language Generation* (NLG) ou Geração de Linguagem Natural. Subárea focada na geração de linguagem natural

Entre as principais aplicações de NLU, destacam-se o *Part-of-Speech* (PoS) *Tagging*, *Named Entity Recognition* (NER), Classificação de Texto como Análise de Sentimento e Modelagem de Tópicos (*Topic Modeling*). Por outro lado, exemplos de NLG envolvem Resumo ou Sumarização de texto (*Text Summarization*), Tradução (*Machine Translation*) e sistemas de *chatbot*.

Entre as abordagens utilizadas para processar texto e gerar inteligência identificadas por meio do mapeamento sistemático descrito na seção 3, destacam-se métodos como reconhecimento de entidades nomeadas (NER), modelagem de tópicos e geração automática de resumos de texto.

No contexto de análise de clientes e concorrentes, a identificação automática de entidades nomeadas (NER), como nomes de empresas, localizações e perfis relacionados, revela-se particularmente útil, e essa técnica permite obter informações valiosas sobre oportunidades de negócios, bem como identificar pontos fortes e fracos de concorrentes, utilizando dados predominantemente provenientes de fontes não estruturadas, como textos de sites (WALTINGER et al., 2013).

Ademais, a modelagem de tópicos desempenha um papel essencial na geração de inteligência estratégica, facilitando a extração de temas predominantes em notícias e permitindo a identificação de tendências emergentes, bem como o acompanhamento de sua evolução ao longo do tempo (ARSLAN; CRUZ, 2022). Por fim, a geração automática de resumos (ATS) destaca-se como uma ferramenta eficiente para auxiliar na tomada de decisão, já que essa técnica possibilita a extração de informações específicas e relevantes a partir de grandes volumes de dados, reduzindo sua extensão sem comprometer a essência das informações (SANCHEZ-GOMEZ; VEGA-RODRíGUEZ; PéREZ, 2022).

2.2 Resumo de Texto Autmomático

Os resumos de texto automáticos podem ser gerados de diversas formas. Quanto ao método, os resumos podem ser abstrativos ou extrativos. Um resumo abstrativo é aquele em que o conteúdo gerado pelo algoritmo é novo, ou seja, as palavras e as sentenças não existem no documento original. Já no resumo extrativo, o algoritmo seleciona um subconjunto de sentenças do documento para compor o resumo (ALGULIYEV; ALIGULIYEV; ISAZADE, 2015; SANCHEZ-GOMEZ; VEGA-RODRíGUEZ; PéREZ, 2020).

Além disso, os resumos podem ser genéricos ou *query-oriented*. Os resumos genéricos não precisam de nenhuma informação do usuário, como o assunto a ser resumido ou temas, e, os algoritmos resumem os documentos baseando-se no contexto geral do(s) texto(s) (SANCHEZ-GOMEZ; VEGA-RODRíGUEZ; PéREZ, 2018), enquanto que no *query-oriented* precisa de alguma informação, principalmente a *query* ou tópico de interesse em forma de sentença, ao fazer um resumo orientado por *queries*, o resumo é feito de acordo com a informação fornecida pelo usuário, focando no tema especificado por ele (HUANG et al., 2010; SANCHEZ-GOMEZ; VEGA-RODRíGUEZ; PéREZ, 2024).

Ademais, os resumos podem ser *single-document* ou *multi-document*. Os métodos *single-document* reduzem a informação contida somente em um documento em um resumo conciso, e os métodos *multi-document* extraem as informações de todos os documentos de um conjunto (SAINI et al., 2019; SANCHEZ-GOMEZ; VEGA-RODRíGUEZ; PéREZ, 2018; MENDOZA et

al., 2014).

As abordagens também podem ser classificadas como supervisionadas e não supervisionadas. A abordagem supervisionada trata o resumo de documentos como um problema de classificação; nele, o modelo de classificação identifica as sentenças que devem ser incluídas no resumo. Mas esses modelos requerem amostras de treinamento. Os métodos não supervisionados utilizam algoritmos de clusterização para pontuar sentenças do documento, a partir de uma combinação de atributos ou *features* predefinidas (ALGULIYEV; ALIGULIYEV; ISAZADE, 2015).

2.3 Algoritmos Bioinspirados

Os métodos de resumo automáticos extrativos são, essencialmente, um problema de otimização combinatória, uma vez que envolvem a seleção de um subconjunto de sentenças a partir de um documento, preservando as informações mais relevantes.

A utilização de algoritmos bioinspirados para a resolução de problemas de otimização justifica-se primordialmente pelas limitações intrínsecas dos modelos tradicionais de aprendizado de máquina e aprendizado profundo, que frequentemente enfrentam obstáculos relacionados à otimização, ao ajuste de parâmetros e à manipulação de dados em larga escala e de alta dimensionalidade (HO et al., 2025).

À medida que problemas computacionais, como os de NLP, tornam-se mais complexos, a exigência por espaços de características de alta dimensionalidade e a necessidade de adaptabilidade em ambientes dinâmicos intensificam-se, tornando as técnicas tradicionais de otimização insuficientes (HO et al., 2025).

Consequentemente, esses algoritmos são ideais para resolver espaços de parâmetros complexos, facilitando o ajuste de hiperparâmetros, a otimização de características e a integração de mecanismos de aprendizado adaptativo (HO et al., 2025).

A computação bioinspirada refere-se ao desenvolvimento e à aplicação de algoritmos baseados em processos biológicos e fenômenos naturais, fundamentando-se em princípios observados na natureza para a resolução de desafios computacionais (KAR, 2016).

Esse campo interdisciplinar estabelece uma interface entre biologia, ciência da computação e matemática, estando fortemente relacionado à inteligência artificial e ao *machine learning*, uma vez que grande parte de suas investigações se insere nesses domínios (SAJJA; AKERKAR, 2013; MAMATHA; JOSHI; AMITH, 2024).

A motivação para o estudo de sistemas biológicos no contexto da computação bioinspirada decorre de características como eficiência, adaptabilidade, robustez, processamento descentralizado e paralelo, além de funcionalidade emergente — propriedades consideradas desejáveis para sua incorporação em sistemas computacionais. Sistemas naturais, em geral, são

compostos por um grande número de entidades de processamento simples, que operam de forma descentralizada, paralela e assíncrona. A funcionalidade global emerge das interações entre esses agentes, conferindo a tais sistemas elevada robustez, paralelismo e capacidade de adaptação a domínios de problemas dinâmicos e em constante transformação.

Nesse contexto, a computação bioinspirada mostra-se particularmente adequada para a resolução de problemas computacionais complexos, dinâmicos e de larga escala, incluindo problemas NP-difíceis, nos quais técnicas tradicionais de otimização matemática podem se tornar ineficazes devido à presença de ótimos locais ou à inviabilidade computacional (SWAYAM-SIDDHA, 2020).

Para enfrentar tais desafios, essa abordagem emprega estratégias evolutivas e mecanismos emergentes, como auto-organização e adaptação, com o objetivo de propor arquiteturas computacionais não convencionais e novos paradigmas de programação. Seu escopo abrange tanto a otimização quanto a modelagem de fenômenos vivos, incluindo aplicações que vão desde ferramentas de otimização global até o aprimoramento de métodos computacionais à medida que a complexidade dos sistemas aumenta.

2.4 Modelos de Linguagem

Quanto os métodos de sumarização de texto abstrativa, os *Large Language Models* (LLMs) ou Grandes Modelos de Linguagem impulsionaram uma mudança de paradigma no Processamento de Linguagem Natural (PLN) (CORRÊA et al., 2024a). Esses modelos demonstraram habilidades emergentes em geração de texto, resposta a perguntas e raciocínio, facilitando tarefas em diferentes domínios (WANG et al., 2025). O campo do PLN foi profundamente transformado pela capacidade dos LLMs de executar tarefas *downstream* após o treinamento em grandes volumes de dados sob um regime de aprendizado auto-supervisionado (CORRÊA et al., 2024a).

Modelos baseados em *transformers*, como BERT, RoBERTa, mT5 e a família de modelos GPT, consolidaram-se como a base para diversas aplicações e linhas de pesquisa em PLN (CORRÊA et al., 2024a). O desenvolvimento de LLMs tem se expandido rapidamente, incluindo tanto modelos proprietários, como ChatGPT, Bard e Claude, quanto modelos de código aberto, como Llama (WANG et al., 2025). Atualmente, grande parte da pesquisa nessa área concentra-se na escalabilidade do tamanho dos modelos, nos dados de treinamento, na eficiência e nas capacidades gerais dessas arquiteturas (CORRÊA et al., 2024a).

Apesar dos avanços, o progresso dos LLMs não tem ocorrido de forma uniforme em todos os idiomas (CORRÊA et al., 2024a). A maioria é treinada em línguas com alta disponibilidade de recursos, como o inglês, enquanto modelos multilíngues apresentam desempenho inferior quando comparados aos monolíngues. Essa disparidade decorre do desequilíbrio nos dados de treinamento, em que idiomas de alto recurso predominam nos corpora, resultando em insatisfação dos usuários quanto às capacidades dos modelos multilíngues em línguas não inglesas (CORRÊA

et al., 2024a).

O uso prático de LLMs ainda enfrenta diversas limitações, como os elevados custos computacionais e regimes de licenciamento restritivos, preocupações com privacidade e segurança dos dados, inviabilidade em dispositivos com menor poder computacional ou de borda (*edge devices*), alta latência de inferência e baixo desempenho em domínios especializados (WANG et al., 2025; CORRÊA et al., 2024a; CORRêA et al., 2025). Uma alternativa para mitigar essas restrições é o uso de *Small Language Models* (SLMs) ou Pequenos Modelos de Linguagem.

Os *Small Language Models* (SLMs) têm recebido crescente atenção como alternativas promissoras aos LLMs (WANG et al., 2025). A definição exata de SLMs pode variar, mas, em geral, são caracterizados por possuírem menos parâmetros do que os LLMs, sendo que algumas classificações consideram modelos com menos de um bilhão de parâmetros (WANG et al., 2025). Outras definições os enquadram como "pequenos"em relação a seus homólogos de maior porte, abrangendo modelos com até 10 bilhões de parâmetros, e destacam a ausência de habilidades emergentes que são típicas de LLMs mais extensos (WANG et al., 2025).

Eles destacam-se por sua baixa latência de inferência, custo-benefício, eficiência no desenvolvimento, além da facilidade de personalização e adaptação (WANG et al., 2025). Eles proporcionam economias computacionais relevantes tanto no pré-treinamento quanto na inferência, com menores demandas de memória e armazenamento, o que é particularmente relevante em aplicações que exigem uso eficiente de recursos (WANG et al., 2025). Tais características tornam os SLMs especialmente adequados para ambientes com restrições de recursos, incluindo dispositivos de borda (*edge devices*) e aplicações em tempo real, nos quais podem contribuir para a melhoria da privacidade, segurança e tempos de resposta por meio do processamento local (WANG et al., 2025; CORRÊA et al., 2024a). Ademais, quando ajustados a domínios específicos, os SLMs podem alcançar, ou até superar, o desempenho de modelos maiores em tarefas especializadas (WANG et al., 2025).

2.5 Classificação de Texto

Métodos baseados em machine learning (ML) são amplamente empregados para prever e medir a corrupção, seja explorando variáveis preditoras em dados tabulares ou identificando anomalias (LIMA; DELEN, 2020; ASH; GALLETTA; GIOMMONI, 2020). As abordagens de ML podem ser divididas em supervisionadas e não supervisionadas. Os métodos supervisionados utilizam amostras de registros previamente rotulados (fraudulentos e não fraudulentos) para construir modelos que classificam novas observações (JOUDAKI et al., 2015).

A detecção automática de corrupção e fraude tornou-se uma área crucial de investigação na gestão pública e nas ciências da computação, impulsionada pelo volume crescente de dados e pela necessidade de aprimorar a fiscalização (AMARAL; RODRIGUES, 2020). A modernização da gestão pública exige a utilização de técnicas que permitam a identificação de padrões

para descobrir ou prevenir atos de improbidade (AMARAL; RODRIGUES, 2020). Diversas metodologias baseadas em inteligência artificial (IA), aprendizado de máquina (machine learning - ML) e mineração de texto (text mining) têm sido propostas para enfrentar este desafio em diferentes contextos, como auditorias governamentais, compras públicas e setor financeiro.

Dentre os métodos supervisionados, O Random Forest (RF), um algoritmo de ensemble, demonstrou ser um dos métodos de classificação mais precisos para prever a percepção de corrupção em uma classificação multiclasse transnacional (atingindo 85,77% de acurácia, superando Máquinas de Vetores de Suporte e Redes Neurais Artificiais) (LIMA; DELEN, 2020). Já Classificadores baseados em árvores, como o Gradient Boosted Classifier (um conjunto de árvores de decisão), têm sido aplicados para prever a presença de corrupção em finanças públicas locais, utilizando dados orçamentários com alta precisão (acurácia de 76%) (ASH; GALLETTA; GIOMMONI, 2020). Enquanto que o Gradient Boosting obteve o melhor desempenho na previsão de intenção de corrupção mesquinha (petty corruption) entre agentes de aplicação da lei, com acurácia acima de 90% (MASROM et al., 2023). Também demonstrou alta acurácia (71%) na classificação de bancos envolvidos em escândalos de corrupção no setor bancário (DAMIANO et al., 2025). Em detecção de fraude em compras públicas, métodos de ensemble (como o Random Forest) superaram outros modelos de ML na detecção de anomalias, conluio e outros tipos de fraude (SANTOS et al., 2025).

A Regressão Logística e a Máquina de Vetores de Suporte (SVM) são amplamente utilizadas para a previsão de índices de corrupção em compras públicas e na detecção de fraude em seguros de saúde (RABUZIN; MODRUŠAN, 2019; JOUDAKI et al., 2015; KOSE; GOKTURK; KILIC, 2015). A Regressão Logística é também utilizada para avaliar o risco de corrupção de servidores públicos, empregando abordagens de imbalanced learning (VASCONCELOS; CHAIM; CAVIQUE, 2021).

Redes Neurais Artificiais (ANN), incluindo o Perceptron Multicamadas (MLP) e arquiteturas de Deep Learning (como LSTM, BiLSTM e CNN), são empregadas em diversas aplicações, como a detecção de fraude em saúde (JOUDAKI et al., 2015; KOSE; GOKTURK; KILIC, 2015) e a classificação de notícias para gestão de riscos de corrupção (WEICHSELBRAUN et al., 2020). Essas arquiteturas avançadas de Deep Learning superaram abordagens clássicas de ML (como Naive Bayes e SVM) na classificação de textos (WEICHSELBRAUN et al., 2020).

Os métodos não supervisionados são úteis para descobrir padrões não visíveis nos dados sem a necessidade de rótulos prévios, sendo essenciais na detecção de anomalias e outliers (AMARAL; RODRIGUES, 2020; JOUDAKI et al., 2015). Técnicas como o Expectation-Maximization (EM) ou K-Means são utilizadas para agrupar atores ou dados com comportamentos semelhantes (KOSE; GOKTURK; KILIC, 2015; JOUDAKI et al., 2015). No contexto de compras públicas, o uso de técnicas de clustering demonstrou resultados superiores na detecção de favoritismo (SANTOS et al., 2025).

Com objetivo de detectar anomalias, algoritmos como Isolation Forest (IF), baseado em

árvores binárias, são eficientes, especialmente em compras públicas, devido ao seu tempo de execução reduzido e requisitos de memória (SANTOS et al., 2025). O PAACDA (Proximity based Adamic Adar Corruption Detection Algorithm) é um algoritmo de aprendizado não supervisionado proposto para detecção abrangente de corrupção em dados tabulares numéricos. Ele utiliza um conceito de algoritmo de grafo (Adamic Adar) para detectar outliers e valores ausentes ou modificados, alcançando alta precisão (99,04% para dados lineares e 96,35% para dados agrupados) (BANNUR et al., 2023). Já o Aprendizado de Máquina Interativo (IML) incorpora o conhecimento de especialistas diretamente no processo de construção de modelos não supervisionados, sendo vital em ecossistemas de fraude dinâmicos onde os padrões de corrupção evoluem rapidamente (KOSE; GOKTURK; KILIC, 2015).

A mineração de texto, do inglês Knowledge Discovery in Text (KDT), é o processo de descoberta de conhecimento potencialmente útil em bases de dados desestruturadas, sendo indispensável dada a grande parte dos dados relevantes estarem em formato textual (AMARAL; RODRIGUES, 2020).

A Análise Textual combinada com algoritmos de machine learning tem sido aplicada para detectar escândalos de corrupção em bancos, analisando relatórios financeiros (DAMIANO et al., 2025).

Uma abordagem inovadora utiliza modelos avançados de linguagem, como o Large Language Model (LLM) FinBERT (uma adaptação de BERT para o domínio financeiro), em conjunto com métodos de dicionário para extrair e analisar o tom (positivo, negativo e contencioso) da divulgação relacionada à governança. Isso é usado como ferramenta preditiva para detectar escândalos de corrupção antes que se tornem públicos (DAMIANO et al., 2025).

O Processamento de Linguagem Natural (NLP) é utilizado para classificar documentos de mídia (como artigos de notícias) quanto ao seu risco de corrupção, empregando algoritmos como Naive Bayes, SVM e arquiteturas de Deep Learning (e.g., CNN, LSTM) (WEICHSELBRAUN et al., 2020). O NLP também é aplicado para extrair informações de documentos de licitação e detectar indicações de corrupção (RABUZIN; MODRUšAN, 2019; SANTOS et al., 2025).

2.6 Modelagem de Tópicos

A Modelagem de Tópicos é uma técnica fundamental de processamento de linguagem natural (NLP) (JIANG et al., 2023; ANGELOV, 2020) e uma poderosa ferramenta não supervisionada utilizada para descobrir a estrutura semântica latente em grandes coleções de documentos, geralmente referida como tópicos (ANGELOV, 2020; GROOTENDORST, 2022).

O objetivo principal da modelagem de tópicos é encontrar descrições curtas dos membros de uma coleção para permitir o processamento eficiente de grandes volumes de dados, enquanto se preservam as relações estatísticas essenciais úteis para tarefas como classificação, detecção de

novidades, sumarização e julgamentos de similaridade (BLEI; NG; JORDAN, 2003), ou seja, descobrir automaticamente os temas (tópicos) latentes que aparecem em um grande conjunto de documentos de texto. Essa técnica é particularmente útil quando o volume de texto é muito grande para ser lido e classificado por uma pessoa (ANGELOV, 2020).

Os modelos de tópicos pressupõem uma estrutura probabilística incluindo tópicos e documentos. Um tópico é o tema, assunto ou matéria de um texto (ANGELOV, 2020). Nos modelos probabilísticos, um tópico é caracterizado por uma distribuição de probabilidade sobre palavras (BLEI; NG; JORDAN, 2003; BLEI; LAFFERTY et al., 2006; TEH et al., 2006). Os tópicos são compartilhados por todos os documentos na coleção (BLEI; LAFFERTY et al., 2006). Um documento é visto como uma mistura aleatória sobre tópicos latentes (BLEI; NG; JORDAN, 2003; TEH et al., 2006). As proporções dos tópicos são específicas para cada documento (BLEI; LAFFERTY et al., 2006).

Historicamente, a modelagem de tópicos evoluiu a partir de métodos de redução de dimensionalidade, como Análise Semântica Latente (LSA) (DEERWESTER et al., 1990), Indexação Semântica Latente Probabilística (pLSA) (HOFMANN, 2013) e Latent Dirichlet Allocation (LDA) (BLEI; NG; JORDAN, 2003). Além disso, variações e novas abordagens surgiram com o tempo, como Fatoração de Matriz Não-Negativa (NMF) (LEE; SEUNG, 1999), Hierarchical Dirichlet Processes (HDP) (TEH et al., 2006) e o BERTopic (GROOTENDORST, 2022).

Esse conjunto de técnicas têm sido aplicadas em uma variedade de aplicações, como no combate a corrupção na Administração Pública.

Motivado pela necessidade de abordar o grande volume de dados textuais e a inserção inadequada de registros, que ocorriam devido ao cadastro descentralizado de novos itens no passado, prejudicando a comparação de valores em novas compras e na identificação de sobrepeços, (AMARAL; RODRIGUES, 2020) utilizou LDA para segmentar dados de auditoria do Governo de Pernambuco (PE) a partir de suas características de suas descrições. O método foi aplicado em um conjunto de 65 mil registros de itens comprados entre 2008 e 2017, procurando disponibilizar informações úteis às ações de controle. A técnica mostrou-se eficaz em detectar tópicos em uma granularidade maior que a classificação humana pré-existente. A identificação de sobrepreços é uma das principais ações de controle executadas pela equipe de auditoria da Secretaria da Controladoria Geral do Estado (SCGE) de PE, prevenindo atos de improbidade entre agentes públicos.

No pico da pandemia de COVID-19, métodos de modelagem de tópico foram utilizadas focadas na saúde pública, com objetivo de compreender os desafios e oportunidades para a vacinação contra a doença analisando posts do Twitter (JIANG et al., 2023)

Historicamente, a modelagem de tópicos evoluiu a partir de técnicas de redução de dimensionalidade, como a Análise Semântica Latente (Latent Semantic Analysis - LSA) (DE-

ERWESTER et al., 1990), a Indexação Semântica Latente Probabilística (Probabilistic Latent Semantic Indexing - PLSA) (HOFMANN, 2013) e a Alocação Latente de Dirichlet (Latent Dirichlet Allocation - LDA) (BLEI; NG; JORDAN, 2003). Ao longo do tempo, diversas variações e extensões foram propostas, incluindo a Fatoração de Matrizes Não Negativas (Non-negative Matrix Factorization - NMF) (LEE; SEUNG, 1999), os Processos Dirichlet Hierárquicos (Hierarchical Dirichlet Processes - HDP) (TEH et al., 2006) e o BERTopic (GROOTENDORST, 2022).

O pLSA, também denominado *Aspect Model*, é uma técnica estatística para a análise de dados de coocorrência ou dados de duas modalidades. Foi introduzido como uma reformulação probabilística da Análise Semântica Latente (LSA) (HOFMANN, 2013). Em contraste com a LSA tradicional — baseada em álgebra linear e na decomposição em valores singulares (SVD) de tabelas de coocorrência — o pLSA fundamenta-se em uma decomposição em misturas derivada de um modelo de classes latentes, oferecendo, assim, uma estrutura estatisticamente mais fundamentada (HOFMANN, 2013).

A base do pLSA reside no *Aspect Model*, um modelo de variáveis latentes para dados de coocorrência. Esse modelo associa uma variável de classe não observada (tópico) $z \in Z = z_1, \ldots, z_k$ a cada observação. Em aplicações textuais, dado um conjunto de documentos $D = d_1, \ldots, d_N$ e um vocabulário $W = w_1, \ldots, w_M$, uma observação de coocorrência corresponde à ocorrência de uma palavra w em um documento d. A matriz de coocorrência $N = (n(d, w))$ registra a frequência com que o termo w ocorre no documento d (representação *bag-of-words*) (HOFMANN, 2013).

A LSA, também conhecida como Indexação Semântica Latente (Latent Semantic Indexing - LSI), é um método de indexação e recuperação automática que busca explorar a estrutura de ordem superior implícita na associação entre termos e documentos, a fim de aprimorar a identificação de materiais relevantes. Essa abordagem visa mitigar deficiências decorrentes dos fenômenos de sinonímia e polissemia (homografia). A LSA trata a imprecisão nos dados de associação termo–documento como um problema estatístico, assumindo a existência de uma estrutura semântica latente subjacente — isto é, um padrão de correlação na coocorrência de palavras entre documentos — parcialmente obscurecida pelo "ruído" da escolha aleatória de palavras. Para estimar essa estrutura semântica latente, a LSA emprega a técnica matemática conhecida como Decomposição em Valores Singulares (Singular Value Decomposition - SVD), modelando o *corpus* como uma matriz retangular termo–documento (DEERWESTER et al., 1990).

O modelo pLSA é estimado por meio da Estimação por Máxima Verossimilhança (Maximum Likelihood Estimation - MLE), utilizando o algoritmo de Expectation–Maximization (EM). Uma das principais vantagens do pLSA é sua capacidade de lidar com a polissemia. O *Aspect Model* pode identificar que um termo polissêmico (como "segment" ou "matrix") pode ser gerado por diferentes fatores latentes (tópicos), dependendo do contexto em que aparece em

um documento. Consequentemente, o termo é explicado por diferentes distribuições condicionais $P(w \mid z)$, resultando em mínima sobreposição na representação fatorada, mesmo quando a palavra ocorre com frequência em diferentes contextos (HOFMANN, 2013).

A LDA é um modelo Bayesiano hierárquico de três níveis, no qual cada item de uma coleção é representado como uma mistura finita sobre um conjunto subjacente de tópicos latentes, e cada tópico é modelado como uma mistura infinita sobre probabilidades de palavras. No contexto da modelagem textual, as probabilidades dos tópicos fornecem uma representação explícita de um documento. A ideia fundamental da LDA é que os documentos são representados como misturas aleatórias de tópicos latentes, sendo cada tópico caracterizado por uma distribuição de probabilidade específica sobre as palavras (BLEI; NG; JORDAN, 2003).

A NMF é um algoritmo desenvolvido para aprender representações baseadas em partes de objetos, como faces, bem como características semânticas de dados textuais. Ela contrasta com outros métodos, como a Análise de Componentes Principais (Principal Component Analysis - PCA) e a Quantização Vetorial (Vector Quantization - VQ), que tipicamente produzem representações holísticas (LEE; SEUNG, 1999). A NMF opera no âmbito da fatoração de matrizes, de forma semelhante à PCA e à VQ, sendo utilizada para modelar a geração de variáveis diretamente observáveis (V) a partir de variáveis latentes (H). O algoritmo busca uma fatoração aproximada da forma $V \simeq WH$, em que V é uma matriz de dados $n \times m$ — por exemplo, uma matriz de contagem de palavras, na qual V_{im} representa o número de ocorrências da i-ésima palavra no m-ésimo documento.

A matriz W é uma matriz $n \times r$ que contém características semânticas, sendo o posto da fatoração (r) escolhido de modo que o produto WH forneça uma representação compacta dos dados em V. A matriz H é uma matriz $r \times m$ cujas colunas, denominadas codificações, consistem nos coeficientes que representam cada documento como uma combinação linear dos vetores base (as colunas de W) (LEE; SEUNG, 1999).

O que distingue a NMF de métodos como a PCA e a VQ é a restrição de não negatividade imposta às matrizes fatoradas, proibindo entradas negativas tanto em W quanto em H. Como todos os elementos não nulos de W e H são positivos, a NMF permite apenas combinações aditivas. A ausência de subtração está alinhada à noção intuitiva de construir um todo a partir de partes, resultando em representações baseadas em partes (LEE; SEUNG, 1999).

A NMF é implementada por meio de regras iterativas de atualização para W e H. A aplicação repetida dessas atualizações converge para um máximo local de uma função objetivo, relacionada à verossimilhança de geração dos dados em V a partir da matriz base W e das codificações H, permitindo que a NMF seja interpretada como um método de modelagem generativa probabilística. Consequentemente, o algoritmo de NMF realiza simultaneamente o aprendizado do conjunto de bases (W) e a inferência das variáveis latentes (H) a partir das variáveis observadas (V) (LEE; SEUNG, 1999).

O HDP é um modelo Bayesiano não paramétrico desenvolvido para problemas de agrupamento envolvendo múltiplos grupos de dados. Ele estende a estrutura do Processo de Dirichlet (Dirichlet Process - DP), que fundamenta modelos de mistura nos quais o número de clusters é indefinido e inferido automaticamente pelo modelo. Cada grupo de dados é modelado como uma mistura cujo número de componentes também é indefinido e determinado durante o processo de inferência. Ademais, esses componentes podem ser compartilhados entre grupos, permitindo capturar dependências de forma eficaz e promovendo a generalização para novos grupos não observados (TEH et al., 2006).

Problemas de agrupamento ocorrem com frequência na prática, como na descoberta de tópicos em corpora de documentos (TEH et al., 2006). A principal vantagem do HDP reside em sua capacidade de inferir automaticamente a complexidade necessária do modelo, ao mesmo tempo em que possibilita a transferência de conhecimento por meio da inferência do número adequado de tópicos, do suporte ao aprendizado por transferência e da manutenção da extensibilidade (TEH et al., 2006).

O BERTopic é uma abordagem de modelagem de tópicos desenvolvida para identificar tópicos latentes coerentes em coleções de documentos, estendendo o paradigma de modelagem de tópicos baseado em agrupamento. Diferentemente de modelos convencionais, como a Alocação Latente de Dirichlet (LDA) e a Fatoração de Matrizes Não Negativas (NMF) — que tratam os documentos como representações *bag-of-words* e desconsideram relações semânticas entre palavras — o BERTopic explora o poder representacional de *embeddings* de texto. Essas representações vetoriais codificam o significado semântico, de modo que textos semanticamente semelhantes são posicionados próximos no espaço vetorial (GROOTENDORST, 2022).

O BERTopic gera *embeddings* de documentos utilizando modelos de linguagem pré-treinados baseados em Transformers, realiza o agrupamento desses *embeddings* e, posteriormente, deriva representações dos tópicos por meio de um procedimento TF-IDF baseado em classes (GROOTENDORST, 2022). O modelo opera em três etapas principais: (1) gera *embeddings* numéricos de alta dimensionalidade dos documentos utilizando o framework Sentence-BERT (SBERT); (2) reduz a dimensionalidade por meio do UMAP e realiza o agrupamento, por padrão utilizando o HDBSCAN, para identificar clusters de documentos semanticamente semelhantes; e (3) extrai representações dos tópicos a partir dos clusters resultantes utilizando o TF-IDF baseado em classes (c-TF-IDF) (GROOTENDORST, 2022).

2.7 Métricas de Avaliação

Nesta Subseção apresentamos as métricas de avaliação adotadas para cada uma das tarefas de Resumo, Classificação de Texto e Modelagem de Tópicos.

2.7.1 Métricas para Avaliação para Resumo de Texto

Nesta Subseção, são descritas as métricas de avaliação adotadas no experimento. Foram utilizadas o ROUGE-N (LIN, 2004), ROUGE-L (LIN, 2004), BLEU (PAPINENI et al., 2002), METEOR (LAVIE; AGARWAL, 2007) e BERTScorcore (ZHANG et al., 2020).

2.7.1.1 ROUGE

A *Recall-Oriented Understudy for Gisting Evaluation* (ROUGE-N) (LIN, 2004), amplamente utilizada na literatura para avaliar a qualidade do resumo gerado por um método de sumarização automática, mensura a qualidade do resumo contando a sobreposição de unidades de sequência de palavras (n-gramas) e pares de palavras entre o resumo automático gerado pelo modelo e o resumo referência (EL-KASSAS et al., 2021). A definição formal é dada pela Eq.2.1.

$$\text{ROUGE-N} = \frac{\sum_{S \in \text{RefSummaries}} \sum_{\text{N-grams} \in S} \text{Count}_{\text{match}}(\text{N-gram})}{\sum_{S \in \text{RefSummaries}} \sum_{\text{N-grams} \in S} \text{Count}(\text{N-gram})} \tag{2.1}$$

Em que $RefSummaries$ representa o conjunto dos resumos de referência, usados para comparação com o resumo gerado automaticamente, item $N - grams$ refere-se aos segmentos consecutivos de N palavras (ou *tokens*) em uma frase ou texto, $\text{Count}_{\text{match}}(\text{N-gram})$ é o número de vezes que um *N-grama* específico do resumo de referência aparece no resumo gerado automaticamente, indicando a contagem de *N-gramas* sobrepostos entre o resumo de referência e o resumo gerado e Count(N-gram) é a contagem total dos *N-gramas* no resumo de referência. A soma no denominador representa todos os *N-gramas* possíveis que poderiam ter sido capturados do resumo de referência.

2.7.1.2 ROUGE-L

A métrica *Recall-Oriented Understudy for Gisting Evaluation – Longest Common Subsequence* (ROUGE-L) (LIN, 2004) é uma medida de avaliação automática de qualidade de textos gerados, fundamentada no cálculo da **maior subsequência comum (Longest Common Subsequence, LCS)** entre um texto de referência R e um texto automático H.

Seja $LCS(R, H)$ o comprimento da maior subsequência comum entre R e H. Define-se *Recall*, *Precision* e Fβ (F1-*score*) como Eq.2.2, Eq.2.3 e Eq.2.4 respectivamente.

em que $|R|$ representa o comprimento da sequência de referência e $|H|$ o comprimento da sequência gerada. Para o cálculo do Fβ, o β geralmente é 1, resultado na métrica F1-*score*.

A utilização da subsequência comum confere ao ROUGE-L a capacidade de capturar a **estrutura global** e a **ordem relativa** das palavras, sem restringir-se à contiguidade estrita exigida por n-gramas. Essa característica diferencia o ROUGE-L do ROUGE-N, permitindo-lhe refletir similaridade textual em um nível mais flexível.

$$\text{R\_LCS} = \frac{LCS(R,H)}{|R|} \tag{2.2}$$

$$\text{P\_LCS} = \frac{LCS(R,H)}{|H|} \tag{2.3}$$

$$\text{F\_LCS} = \frac{(1+\beta^2)\cdot R\_LCS \cdot P\_LCS}{R\_LCS + \beta^2 \cdot P\_LCS} \tag{2.4}$$

2.7.1.3 BLEU

A métrica Bilingual Evaluation Understudy (BLEU) (PAPINENI et al., 2002) quantifica a qualidade de um texto gerado por meio da **precisão de n-gramas** em relação a uma ou mais referências, incorporando uma penalidade de brevidade para evitar resumos automáticos excessivamente curtas. Formalmente, seja p_n a precisão de n-gramas de ordem n, com pesos w_n (usualmente uniformes) e BP a penalidade de brevidade. Define-se:

$$\text{BLEU} = BP \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \tag{2.5}$$

$$BP = \begin{cases} 1 & \text{if } |H| > |R| \\ e^{(1-|R|/|H|)} & \text{if } |H| \leq |R| \end{cases}$$

2.7.1.4 METEOR

A métrica *Metric for Evaluation of Translation with Explicit ORdering* (METEOR) (LAVIE; AGARWAL, 2007) estabelece uma correspondência flexível entre resumos automáticos e referências, considerando coincidências exatas, radicais, sinônimos e paráfrases. Define-se a precisão P e o *Recall* R sobre as correspondências identificadas. A pontuação F_α é calculada conforme a Eq. 2.6. O α pondera a importância relativa entre precisão e *Recall*. Uma penalidade de fragmentação Pen, dependente da dispersão dos segmentos correspondentes. Por fim, METEOR é calculada conforme 2.7.

$$F_\alpha = \frac{PR}{\alpha P + (1-\alpha)R} \tag{2.6}$$

$$\text{METEOR} = (1 - Pen) \cdot F_\alpha \tag{2.7}$$

2.7.1.5 BERTScore

A métrica BERTScorcore (ZHANG et al., 2020) fundamenta-se em representações semânticas obtidas por modelos de linguagem pré-treinados baseados em Transformers. Dado um

conjunto de embeddings $\{e_h\}$ para tokens do resumo automático e $\{e_r\}$ para tokens da referência onde $\cos(e_h, e_r)$ é a similaridade de cosseno entre os embeddings. Diferente das demais métricas, o BERTScore captura a semântica real, incluindo sinônimos e paráfrases, e possui uma correlação com a avaliação humana porque a representação numérica do texto de referência e do resumo automático é contextual, por meio de embeddings. BERTScore é calculada conforme 2.10

$$P = \frac{1}{|H|} \sum_{h \in H} \max_{r \in R} \cos(e_h, e_r) \tag{2.8}$$

$$R = \frac{1}{|R|} \sum_{r \in R} \max_{h \in H} \cos(e_r, e_h) \tag{2.9}$$

$$\text{F\_BERT} = \frac{2PR}{P + R} \tag{2.10}$$

2.7.2 Métricas de Avaliação para Classificação de Texto

As seguintes métricas de avaliação foram utilizadas para analisar os resultados de classificação: *Accuracy*, *Precision*, *Recall* e F1-*score* (ZHU; ZENG; WANG, 2010). A Tabela 1 apresenta a definição e o cálculo de cada métrica. Essas métricas baseiam-se nas frequências de Verdadeiro Positivo (TP), Verdadeiro Negativo (TN), Falso Positivo (FP) e Falso Negativo (FN), que representam:

- ***True Positive* (TP)**: Número total de instâncias positivas (notícias) corretamente classificadas como positivas.
- ***True Negative* (TN)**: Número total de instâncias negativas (notícias) corretamente classificadas como negativas.
- ***False Positive* (FP)**: Número total de instâncias negativas (notícias) incorretamente classificadas como positivas.
- ***False Negative* (FN)**: Número total de instâncias positivas (notícias) incorretamente classificadas como negativas.

2.7.3 Métricas de Avaliação para Modelagem de Tópicos

As seguintes métricas de avaliação foram utilizadas para avaliar a qualidade e a interpretabilidade semântica dos tópicos gerados: **Coherence Value (C_V)** e **Normalized Pointwise Mutual Information (C_NPMI)** (RÖDER; BOTH; HINNEBURG, 2015). Essas métricas visam quantificar a similaridade semântica entre as palavras mais representativas dentro de cada tópico, refletindo o grau em que as palavras tendem a coocorrer no *corpus* de referência.

Tabela 1 – Métricas de Avaliação para Classificadores

Métrica	**Descrição**	**Fórmula**
Accuracy	Representa a porcentagem de instâncias (notícias) corretamente classificadas, considerando tanto verdadeiros positivos quanto verdadeiros negativos.	accuracy = $\frac{TP+TN}{TP+TN+FP+FN}$
Precision	Mede a proporção de instâncias previstas como positivas que realmente pertencem à classe positiva, isto é, a confiabilidade das predições positivas.	precision = $\frac{TP}{TP+FP}$
Recall	Também denominada sensibilidade, mede a proporção de instâncias positivas que foram corretamente identificadas pelo modelo.	recall = $\frac{TP}{TP+FN}$
F1-*score*	Média harmônica entre precisão e sensibilidade, visando equilibrar ambos os indicadores; especialmente útil em cenários com classes desbalanceadas.	F1-*score* = $2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$

2.7.3.1 Coherence Value (C_V)

***Coherence Value* (C_V)** é uma métrica desenvolvida para maximizar a correlação com avaliações humanas. O C_V baseia-se em uma janela deslizante aplicada aos textos de referência, mensurando o grau de consistência semântica entre as palavras mais representativas de um tópico, considerando sua coocorrência e relações contextuais. Essa métrica é particularmente eficaz para correlacionar avaliações automatizadas com julgamentos de interpretabilidade humana.

O cálculo é realizado por meio de uma janela deslizante sobre o *corpus* de referência, a fim de estimar probabilidades de coocorrência e construir vetores de contexto para cada palavra. A similaridade semântica entre duas palavras (w_i) e (w_j) é então quantificada utilizando a similaridade do cosseno (sim) entre seus vetores de contexto. A coerência geral de um tópico é computada como a média das similaridades entre todos os pares de palavras (P), conforme apresentado na Equação 2.11:

$$\text{C\_V} = \frac{1}{|P|} \sum_{(w_i, w_j) \in P} \text{sim}(w_i, w_j) \tag{2.11}$$

2.7.3.2 Normalized Pointwise Mutual Information (C_NPMI)

***Normalized Pointwise Mutual Information* (C_NPMI)** é derivada da *Pointwise Mutual Information* (PMI) e quantifica a dependência estatística entre pares de palavras em um tópico. A normalização restringe o valor ao intervalo ([-1, 1]), no qual valores mais elevados indicam associações positivas mais fortes entre as palavras, refletindo maior coerência temática; em outras palavras, indica que as palavras coocorrem com maior frequência do que seria esperado ao acaso, evidenciando uma coerência semântica mais robusta no interior do tópico. Para duas palavras (w_i) e (w_j), com probabilidade conjunta $p(w_i, w_j)$ e probabilidades individuais $p(w_i)$ e $p(w_j)$, o NPMI é definido como:

$$\text{NPMI}(w_i, w_j) = \frac{\log \frac{p(w_i, w_j)}{p(w_i)p(w_j)}}{-\log p(w_i, w_j)} \tag{2.12}$$

2.8 Avaliação Experimental

A condução de uma experimentação é um processo complexo, como destacado por Wohlin et al. (2024), envolvendo a preparação, execução e análise meticulosa dos experimentos. Travassos, Gurov e Amaral (2020) enfatiza que a experimentação é o cerne do processo científico, fornecendo um método sistemático e controlado para avaliar as atividades humanas.

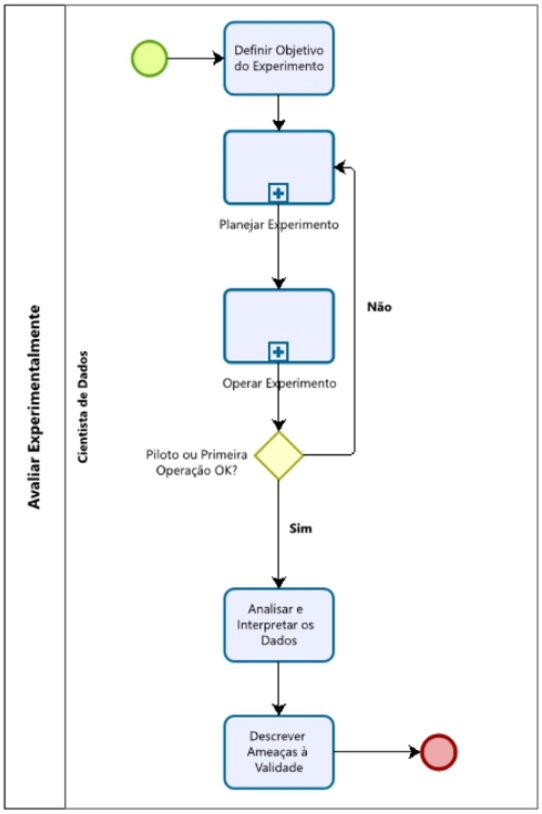
Os experimentos controlados seguem os passos propostos por Colaço JÚNIOR (2025), que consistem na Definição do Objetivo, Planejamento, Operação, Análise e Interpretação dos Dados e Descrição de Ameaças à Validade. A etapa de Planejamento é composta pela subetapa de Seleção de Contexto, Definição de Questões de Pesquisa, Definição de Variáveis Dependentes e Independentes, Formulação de Hipóteses Teóricas, Seleção de Participantes e/ou Objetos, Projeção do Experimento e Definição de Instrumentação. O Processo de Operação de Experimento se desdobra em Preparação, Execução do Experimento e Definição e Execução de Validação dos Dados. A Figura 1a representa as etapas de Avaliação Experimental. Enquanto que as Figuras 1b e 2 descrevem os subprocessos das etapas de Planejamento e Operação do Experimento. Cada Avaliação Experimental é descrita em seu respectivo capítulo.

2.9 Health News Related Dataset

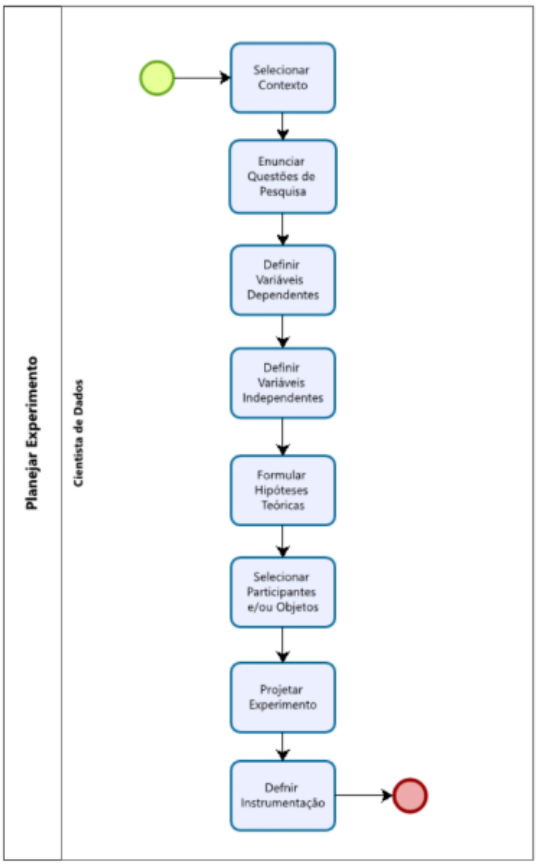
Esta seção descreve o processo de criação da base de dados utilizada na avaliação dos experimentos controlados nos Capítulos 7, 8 e 9.

A construção da base de dados, incluindo todas as etapas para coleta e armazenamento das informações, está fora do escopo deste trabalho. Esta Seção descreve o processo realizado por Fontes et al. (2023), que foi executado em três etapas. A primeira etapa consistiu em uma prova de conceito baseada em entrevistas com os auditores da Auditoria do Sistema Único de Saúde (AudSUS), com o objetivo de esclarecer o processo de captação do material utilizado na fase analítica da auditoria. A fase analítica, primeira etapa de uma auditoria, corresponde ao planejamento e visa preparar a equipe para a fase operativa, proporcionando uma compreensão mais detalhada do contexto das atividades subsequentes e auxiliando na construção do conhecimento necessário sobre o objetivo da auditoria.

A segunda etapa foi um estudo exploratório, cujo objetivo era mapear como as matérias textuais estão organizadas nas fontes, identificar sua rotina de publicação, periodicidade, limitações existentes e possíveis soluções para casos de dados ausentes ou incompletos. Com a conclusão desse mapeamento, iniciou-se a construção de um modelo responsável pelo armazenamento desses dados.



(a) Atividades do Processo Avaliar Experimentalmente (Colaço JÚNIOR, 2025).



(b) Atividades do Subprocesso Planejar Experimento (Colaço JÚNIOR, 2025).

Figura 1 – Processo e Subprocesso de Avaliação Experimental

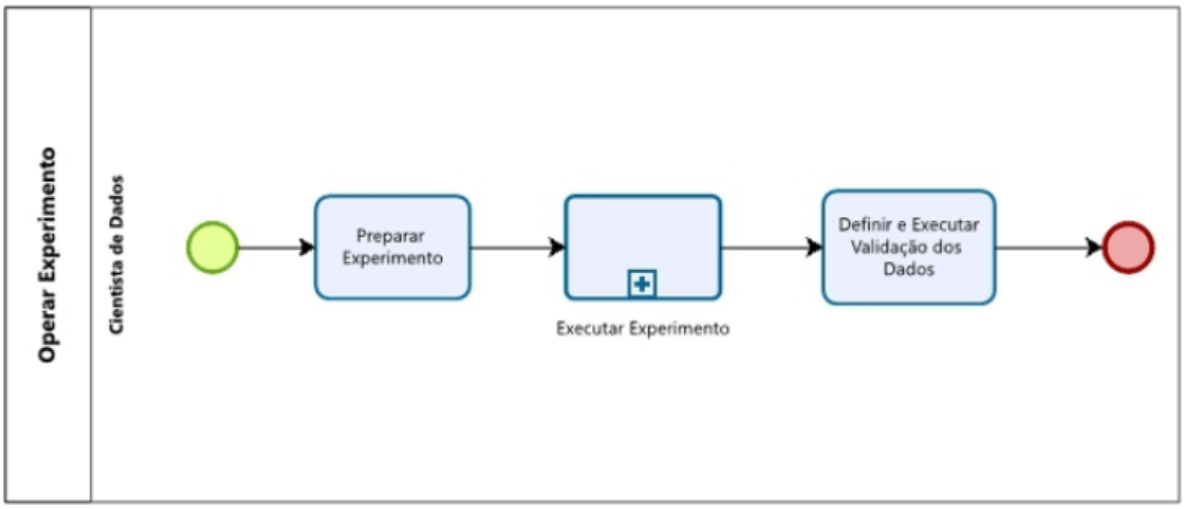


Figura 2 – Atividades do Subrocesso Operar Experimento (Colaço JÚNIOR, 2025).

Na terceira etapa, foi realizada a construção da base de dados. Para isso, Fontes et al. (2023) desenvolveram um fluxo de coleta utilizando a linguagem de programação Python, o framework Django, o banco de dados PostgreSQL e o OpenSearch para indexação dos resultados. A coleta das notícias foi efetuada com robôs especializados, capazes de ler, interpretar e coletar os

metadados das fontes. A configuração desses robôs permite que coletem apenas novas matérias ou reprocessem publicações anteriores. Essa funcionalidade possibilita buscas mais frequentes em fontes que publicam várias matérias ao longo do dia, em contraste com aquelas que realizam apenas uma publicação diária, mensal ou ocasional, como no caso de relatórios de auditoria e diários oficiais. A base de dados final contém mais de 6 milhões de matérias textuais provenientes de fontes de pesquisa indicadas pelos auditores consultados na fase inicial.

Após a coleta, as matérias passam por um fluxo de pré-processamento, responsável por higienizar os textos e padronizar seus metadados de acordo com o modelo estabelecido. Esse processo é fundamental para garantir a qualidade dos dados e seu correto armazenamento no banco de dados e na ferramenta de indexação.

2.9.1 Curadoria Base de Dados

As informações coletadas por Fontes et al. (2023) foram organizadas numa base de dados contendo 154.407 notícias com informações sobre a data de publicação, fonte (site), título da notícia, resumo (*headline*) e conteúdo da notícia. Todo o processo de manipulação da base de dados foi executado utilizando a ferramenta Jupyter Notebook[1] por meio da biblioteca Polars[2] na versão 1.32.0 utilizando a linguagem de programação Python [3] na versão 3.10.12.

Para identificar as notícias relacionadas à saúde com indícios de auditoria, uma estratégia de seleção em três etapas baseada em palavras-chave foi executada. Na primeira etapa, utilizando toda a base de dados, as notícias que possuíam a palavra-chave "saúde"em seu **conteúdo** (*corpus*) foram classificadas como "Saúde"e as demais foram classificadas como "Notícia Genérica". As notícias genéricas foram desconsideradas das próximas etapas. Nessa primeira etapa, 9516 notícias foram classificadas como "Saúde". Enquanto que as 144891 notícias restantes foram desconsideradas das etapas seguintes, pois eram notícias genéricas.

Na segunda etapa, foram utilizadas palavras-chave que indicassem indícios de irregularidade no conjunto de notícias classificadas previamente como "Saúde", dividindo o grupo entre notícias que continham ou não essas palavras-chave, respectivamente "Saúde Genérica"e "Saúde Irregularidade". As notícias que continham pelo menos uma das palavras-chave no título e/ou resumo (*headline*) foram classificadas como "Notícia Irregularidade". Nessa segunda etapa, 6239 notícias com indício de irregularidade foram identificadas, enquanto que as 3277 demais foram classificadas como "Saúde Genérica". A Tabela 2 apresenta as palavras-chave utilizadas.

O processo de correspondência de palavras-chave adotado baseou-se na identificação da presença de termos previamente radicalizados por meio da técnica de *stemming*.

Na terceira etapa, os artigos pertencentes ao subgrupo "Notícia Irregularidade"foram avaliados de forma independente por dois anotadores. Observou-se um acordo interavaliadores

[1] https://jupyter.org/
[2] https://docs.pola.rs/
[3] https://www.python.org/

Tabela 2 – Palavras-chave utilizadas para identificar sinais de irregularidade.

Keywords
abuso, abuso de poder, acordo ilegal, acusaç, acusação, apropriação indébita, auditoria, aumento orçamento, bilhões, cartel, coação, compra, compras públicas, conluio, contrato, contratos, corrupto, corrupção, corte orçamento, crime, crime organizado, criminoso, deflagrou, denuncia, denúncia, desassistência, desfalque, desonestidade, desonesto, desperdício, desvio, desvios, disfarce, documento alterado, dolo, enganar, engano, enganos, enganoso, enriquecimento, enriquecimento ilícito, escândalo, esquema, evasão, falcatrua, falsa declaração, falsificado, falsificador, falsificação, falso, falta, falta de equipamentos, falta equipamento, fiscalização, forjar, fraudador, fraudar, fraude, fraude em contratos, fraude em licitações, fraude financeira, fraude licitação, fraudulento, fugir, golpe, ilegal, ilusão, ilícito, indicativo, indício, infração, investiga, investigação, irregular, irregularidade, irregularidade administrativa, irregularidade de gestão, irregularidade financeira, irregularidades, lavagem, lavagem de dinheiro, licitação, mandado, manipulado, manipulador, manipulação, manipulação de dados, maquiar, mentira, milhões, má conduta, negligência dolosa, ocultar, ocultação, PF, peculato, perjúrio, plano, plano de saúde, plano saúde, prevaricação, propina, recurso, relatório falso, rombo, roubo, sem autorização, sem consentimento, sobrefaturamento, sonegar, sonegação, suborno, sugestão, superfaturamento, suspeit, suspeita, suspeito, transação, transação suspeita, transgressão, transparência, uso indevido, uso indevido de recursos, plano saude, uso irregular, venda.

substancial ou *Inter-Rater Agreement* (IRA) (LANDIS; KOCH, 1977), com valor do Kappa de Cohen (k) igual a 0.6203. A Tabela 3 apresenta a matriz de contingência das avaliações.

Cinco avaliadores adicionais foram designados para resolver os casos de discordância na classificação. Cada avaliador foi individualmente responsável por tomar a decisão final sobre 65 amostras distintas. A avaliação consistiu na análise dos títulos e resumos de todos os artigos, a fim de confirmar sua categorização como *"Health Irregularity"*, *'Generic Health"* ou *"Generic News"*.

Tabela 3 – Matriz de contingência das anotações entre o Avaliador 1 e o Avaliador 2.

Avaliador 1 \Avaliador 2	**Generic News**	**Health (Generic)**	**Health (irregularity)**	**Total Linha (R_i)**
Generic News	4532	302	145	4979
Health (generic)	146	676	7	829
Health (irregularity)	211	17	203	431
Total (C_i)	**4889**	**995**	**355**	$N = 6239$

Por fim, após a terceira etapa, foram identificados 421 artigos relacionados à saúde com indícios de irregularidade. A Figura 3 ilustra o processo de seleção das notícias.

Para a utilização deste conjunto de dados na tarefa de sumarização de textos, foi necessário elaborar resumos dos artigos originais a fim de servirem como resumos de referência para a avaliação daqueles gerados automaticamente. Esses resumos de referência foram produzidos por um pesquisador jornalista externo. Por fim, disponibilizamos as bases de dados originais e anotadas publicamente no repositório Zenodo (GUIMARÃES et al., 2025).

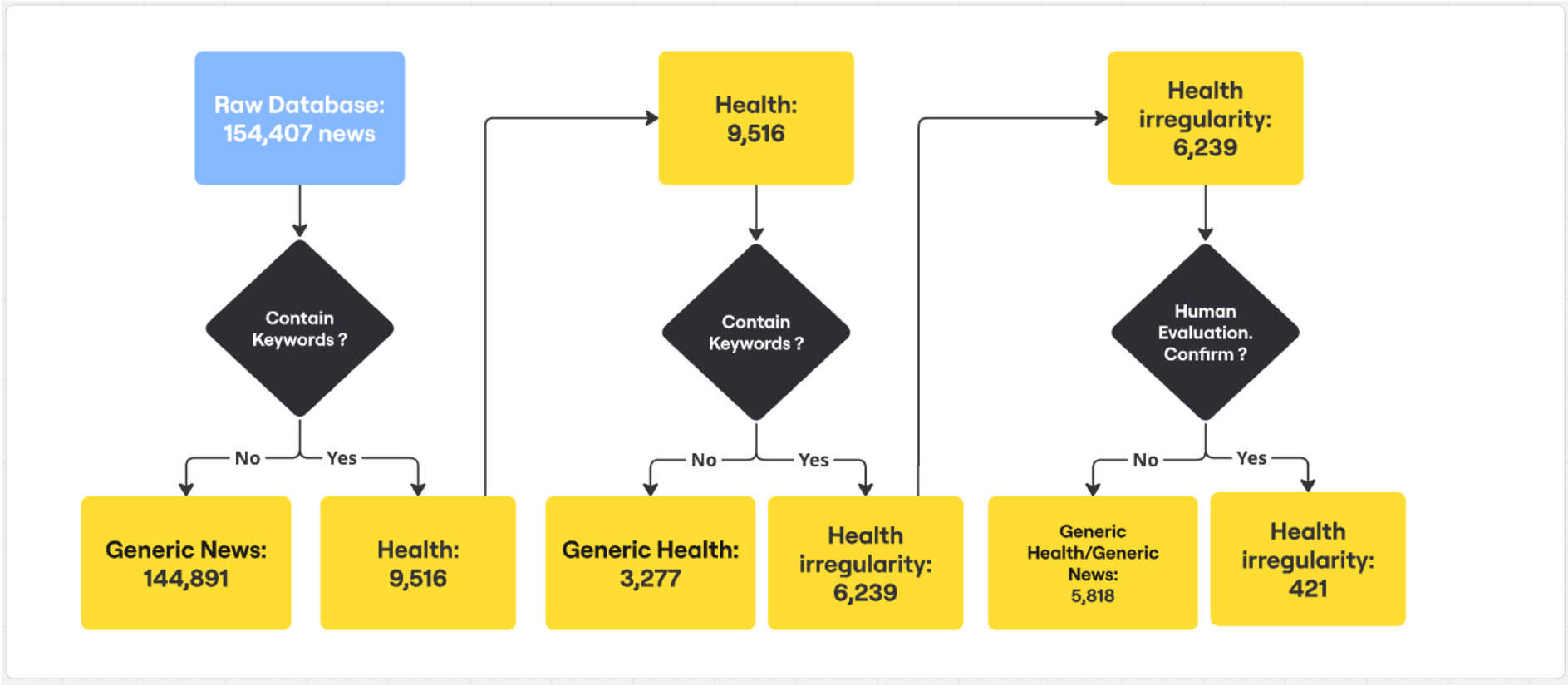


Figura 3 – Processo de Seleção de Notícias de Saúde.

3

Mapeamento Sistemático da Literatura

Este capítulo apresenta parcialmente o mapeamento sistemático da literatura (MSL) intitulado *A Systematic Mapping of Text Summarization Methods Applied in Health Domain*, submetido para o XXII Simpósio Brasileiro de Sistemas de Informação (SBSI), o qual será a base bibliográfica para a realização deste trabalho.

3.1 Materiais e Métodos

A seção a seguir descreve a metodologia empregada neste estudo. Para orientar a formulação da questão de pesquisa e a busca bibliográfica, foi utilizada a estratégia PICO (KITCHENHAM, 2004). A estratégia PICO direciona a construção de questões de pesquisa e buscas bibliográficas, permitindo que os pesquisadores localizem, com precisão e agilidade, as melhores evidências científicas disponíveis. Essa estratégia é composta por quatro elementos fundamentais da pesquisa: Population, Intervention, Control e Outcome, os quais são utilizados para descrever todos os componentes relacionados ao problema identificado e para estruturar as questões de pesquisa.

O objetivo desta pesquisa é investigar os estudos mais recentes sobre sumarização de textos no domínio da saúde, por meio da análise dos métodos desenvolvidos, dos desafios enfrentados, dos benefícios alcançados e das áreas de aplicação.

As revisões sistemáticas da literatura são concebidas para sintetizar estudos existentes sobre um determinado tema, identificar lacunas no conhecimento atual e revelar tendências dominantes na área de interesse. Essa abordagem estruturada possibilita a formulação de resultados imparciais e reprodutíveis, ao tratar as questões de pesquisa por meio de um processo objetivo e metodológico. Foram consideradas diretrizes provenientes da área de engenharia de software, em razão da perspectiva da ciência da computação adotada neste trabalho.

Embora originalmente elaboradas para a engenharia de software, as diretrizes propostas

por Kitchenham (2004) baseiam-se em princípios oriundos da pesquisa em saúde e adaptam metodologias rigorosas comumente empregadas em estudos médicos. Adicionalmente, foram incorporados elementos metodológicos relevantes do checklist PRISMA (PRISMA Executive, 2024), amplamente utilizado como padrão de relato para revisões sistemáticas. O PRISMA é particularmente reconhecido por sua aplicabilidade na avaliação de intervenções em saúde (PAGE et al., 2021).

Dessa forma, com base nas diretrizes propostas por Kitchenham (2004), esta revisão adota as seguintes etapas:

- Definir e descrever as Questões de Pesquisa;
- Definir e descrever a Estratégia de Busca;
- Definir e descrever os Critérios de Seleção das Fontes;
- Definir e descrever a Estratégia de Extração das Informações;
- Definir e descrever a Avaliação da Qualidade.

3.1.1 Questões de Pesquisa

O desenvolvimento das questões de pesquisa teve como objetivo fornecer uma visão geral do campo, com foco nos aspectos fundamentais dos estudos primários (KITCHENHAM, 2004). Essas questões buscam evidenciar os efeitos da intervenção sobre uma população específica e estruturar a pesquisa em quatro elementos fundamentais: Population, Intervention, Control e Outcomes. A Tabela 4 ilustra o modelo PICO utilizado neste estudo.

Tabela 4 – Categorias da Estratégia PICO.

Categoria	**Descrição**
Population	Publicações que abordam o desenvolvimento, a implementação ou o uso de sumarização automática de textos no domínio da saúde
Intervention	Métodos, técnicas e abordagens para sumarização automática de textos
Control	-
Outcomes	Apoio e aprimoramento da tomada de decisão clínica e diagnóstica, bem como a redução da sobrecarga informacional

Dada a definição do modelo PICO e com base nas diretrizes do Protocolo de Mapeamento Sistemático da Literatura (KITCHENHAM, 2004), as questões de pesquisa (RQ) foram definidas da seguinte forma:

- RQ1 - Quais são os principais métodos aplicados para a geração de resumos automáticos de texto no contexto da saúde?
- RQ2 - Em quais áreas da saúde as técnicas de sumarização de textos são aplicadas?

- RQ3 - Em quais anos foi publicado o maior número de artigos nessa área?
- RQ4 - Quais países possuem publicações nessa área?

3.1.2 Estratégia de Busca

As principais bases de dados responsáveis pela publicação dos periódicos mais relevantes nas áreas de Computer Science, Medicine e Biomedicine foram selecionadas para a execução da estratégia de busca dos artigos. As bases de dados foram escolhidas utilizando Wang et al. (2021) e Mishra et al. (2014) como estudos de controle. As bases selecionadas foram ACM Digital Library, IEEE Digital Library, Web of Science, PubMed e Scopus. O período de busca compreendeu de 01/01/2010 a 30/11/2024, e a busca foi realizada em 12/12/2024. Em seguida, foram utilizados os mecanismos de filtragem disponíveis em cada base para considerar título, resumo e palavras-chave. As bases de dados foram acessadas por meio do Portal de Periódicos da CAPES (Capes, 2024), via assinatura institucional.

Para a busca nas bases de dados, foi definida uma string de busca, composta por termos em inglês, seus sinônimos e termos relacionados. Os termos foram identificados com base nas categorias do modelo PICO, definidas na Tabela 4, e posteriormente refinados para melhorar a efetividade da string. Os termos por categoria são apresentados na Tabela 5. As palavras-chave foram selecionadas a partir de artigos de controle semelhantes à solução buscada neste trabalho.

Com base nesses termos, a string de busca apresentada na Tabela 6 foi desenvolvida por meio do ajuste das palavras-chave.

Tabela 5 – Termos por Categoria.

Categoria	**Descrição**
Population	medical, biomedical, biomedicine, health, patient care, healthcare, clinical, disease, therapy, therapies, treatment, diagnosis, diagnoses, etiology, medical texts summarization, clinical texts summarization, biomedical texts summarization, medical domain, biomedical domain, bioinformatics, biomedical literature, medical news, biomedical news.
Intervention	automatic text summarization, ATS, text mining, text summarization, extractive summarization, abstractive summarization, single document summarization, multi-document summarization, generic summarization, query-based summarization.
Control	-
Outcomes	clinical decision support, decision-making improvement, enhanced clinical decisions, improved diagnostic accuracy, optimized treatment planning, improved health delivery, information overload reduction, data simplification, concise clinical summaries, reduction in cognitive load, improved information retrieval efficiency, streamlined clinical workflows

3.1.3 Critérios de Seleção de Fontes

Os critérios de inclusão e exclusão foram definidos com o objetivo de filtrar os artigos relevantes para o mapeamento sistemático. A busca foi conduzida utilizando a string apresentada na Seção 3.1.2, e apenas os estudos selecionados para avaliação foram considerados, ou seja, aqueles que atenderam aos critérios de inclusão (IC) e exclusão (EC). A Tabela 7 descreve os critérios adotados.

Tabela 6 – Termos por Categoria Ajustados.

Categoria	**Descrição**
Population	("health"OR "bioinformatic*"OR "biomed*"OR "clinical"OR "diagnos*"OR "disease*"OR "etiolog*"OR "healthcare*"OR "medical*"OR "patient care"OR "therap*"OR "treatment*"OR "biomed* domain*"OR "biomed* literature"OR "biomed* news*"OR "biomed* text summar"OR "clinical text* summar"OR "medical domain"OR "medical news"OR "medical texts summar*") AND
Intervention	("automatic text summar*"OR "ATS"OR "abstractive summar*"OR "extractive summar*"OR "generic summar*"OR "multi-document summar*"OR "query-based summar*"OR "single document summar*"OR "text mining"OR "text summarization*") AND
Control	-
Outcomes	("clinical decision support"OR "concise clinical summar*"OR "data simplification"OR "decision-making improv*"OR "enhanc* clinical decision*"OR "improv* diagnos* accuracy"OR "improv* healthcare delivery"OR "improv* information retrie* efficien*"OR "information overload reduc*"OR "optimi* treatment plan*"OR "reduc* in cognitive load"OR "streamli* clinical workflow")

Tabela 7 – Critérios de Inclusão e Exclusão.

Critério	**Descrição**
IC1	Estudos com foco nos contextos médico, biomédico, ciências da saúde e informática em saúde.
IC2	Estudos publicados em periódicos ou anais de conferências, nos quais os termos de busca aparecem no título ou no resumo.
IC3	Estudos que descrevem claramente os componentes (por exemplo, métricas) e os métodos utilizados.
EC1	Estudos não redigidos em inglês.
EC2	Estudos duplicados.
EC3	Estudos que não se concentram nos contextos médico, biomédico, ciências da saúde ou informática em saúde.
EC4	Estudos secundários, tais como revisões de literatura, mapeamentos, surveys ou meta-análises.
EC5	Estudos indisponíveis para download ou acesso.
EC6	Estudos que não têm como foco a sumarização de textos no contexto médico.

3.1.4 Estratégia de Extração de Informações

Para responder às Questões de Pesquisa (RQ) mencionadas na Seção 3.1.1, as questões apresentadas na Tabela 8 foram elaboradas e utilizadas na etapa de extração de informações. As questões 1 e 4 foram de múltipla escolha; as questões 2 a 5 foram de escolha única; enquanto as questões 6 e 7 foram abertas.

Tabela 8 – Formulário de Extração de Dados.

Descrição	**Valores**
1. Quais métodos são aplicados?	[Mathematical/Statistics, *machine learning* (ML), *Deep Learning* (DL), *Computational Linguistics* (CL)]
2. Qual é o propósito da sumarização?	[Generic Summarization, Query-oriented Summarization]
3. Qual é o tipo de entrada?	[Multi-Document, Single Document]
4. Quais fontes de dados foram utilizadas?	[Literature, EHR/MDR, News, Others]
5. Qual é o tipo de saída?	[Abstractive, Extractive]
6. Em qual área o método foi aplicado?	
7. Quais países possuem publicações sobre o tema?	

3.1.5 Avaliação de Qualidade

Para assegurar a inclusão de estudos de alta qualidade e identificar e excluir aqueles com potenciais falhas metodológicas, a qualidade dos artigos selecionados foi avaliada com base nos critérios descritos na Tabela 9. Para cada questão do checklist, foram disponibilizadas três opções de resposta: “Yes”, “Partially” e “No”, correspondendo aos pesos 1, 0,5 e 0, respectivamente. Cada artigo foi avaliado por meio da resposta a todas as questões do checklist, e a pontuação total de qualidade foi obtida pela soma dos pesos atribuídos. Estabeleceu-se um limiar mínimo de 3 pontos como requisito para inclusão. Todos os artigos avaliados atingiram pontuação igual ou superior a 3; portanto, nenhum foi excluído do estudo.

Tabela 9 – Checklist de Avaliação da Qualidade

Questão	**Descrição**
Q1	Os objetivos da pesquisa estão claramente definidos?
Q2	O estudo aborda aplicações práticas ou aplica uma metodologia apropriada?
Q3	O estudo é relevante para o tema da pesquisa?
Q4	A técnica proposta é descrita de forma clara?
Q5	Há discussão suficiente dos trabalhos relacionados, incluindo comparações com técnicas concorrentes?

3.2 Condução do Mapeamento Sistemático

A base de dados Scopus foi escolhida como referência para a definição e o refinamento da string de busca. Após ser definida e refinada, a string foi utilizada nos mecanismos de busca das demais bases de dados consideradas neste estudo. O período de busca compreendeu de 01/01/2010 a 30/11/2024, sendo a execução realizada em 17/03/2025. Considerando que métodos tradicionais ainda são amplamente adotados, esse intervalo de 15 anos foi definido com o objetivo de incluir tanto os métodos mais influentes quanto os mais recentes. Ao todo, foram recuperados 447 artigos, sendo 102 (22,82%) da ACM, 20 (4,47%) da IEEE, 54 (12,08%) da Web of Science, 195 (43,63%) da PubMed e 76 (17,00%) da Scopus. A distribuição dos artigos por base de dados é apresentada na Figura 4.

As etapas de seleção dos artigos foram conduzidas por meio da plataforma parsif.al (Parsifal, 2025). Após a realização das buscas nas bases de dados e a importação dos estudos para a plataforma, iniciou-se o processo de filtragem. Inicialmente, todos os artigos duplicados foram classificados como rejeitados, sendo identificados, nessa etapa, 95 artigos (21,25%) como duplicatas. Em seguida, o processo de filtragem prosseguiu de acordo com os critérios de inclusão e exclusão definidos na Seção 3.1.3, e cada artigo foi classificado como Aceito ou Rejeitado após a leitura do título, das palavras-chave e do resumo. Nessa segunda etapa, 319 artigos (71,36%) foram classificados como Rejeitados e 33 (28,64%) como Aceitos. A terceira etapa da seleção consistiu na leitura do texto completo dos estudos, na qual 10 artigos (2,24%) foram classificados como Rejeitados. Ao final, 23 artigos (5,15%) foram selecionados. O fluxograma apresentado na

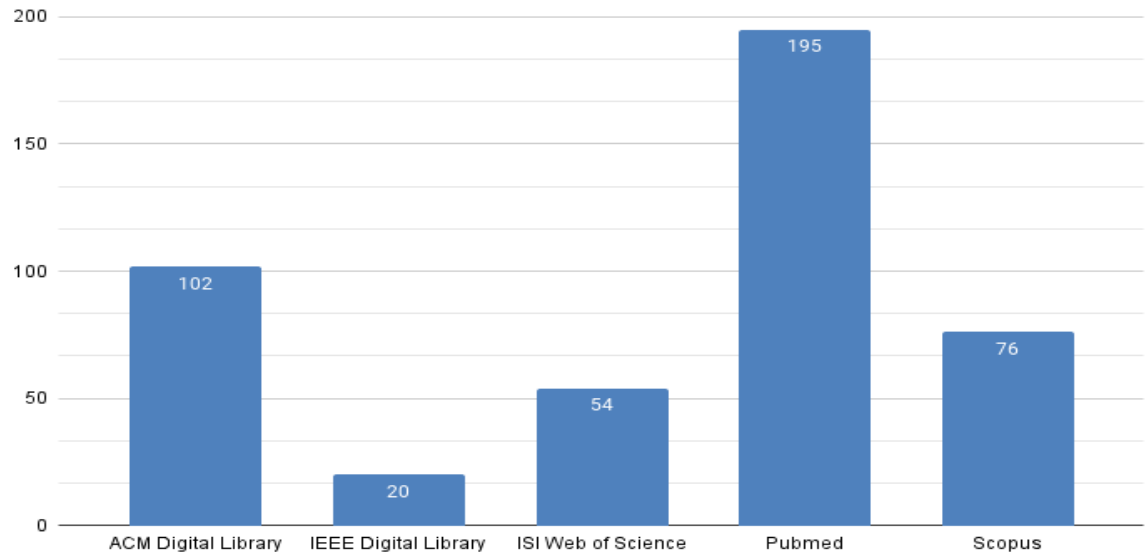


Figura 4 – Distribuição dos artigos por base de dados.

Figura 5 descreve o processo de extração dos artigos obtidos, bem como as exclusões realizadas com base nos respectivos critérios.

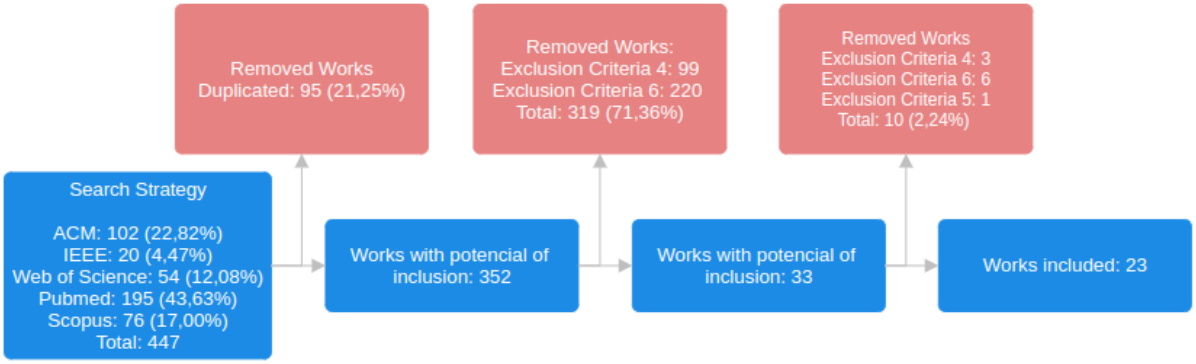


Figura 5 – Processo de extração de dados. Gráfico PRISMA adaptado.

3.3 Resultados

Esta seção apresenta os resultados do mapeamento sistemático, abordando as questões de pesquisa respondidas com base nos dados extraídos por meio do formulário de extração.

3.3.1 RQ1 - Quais são os principais métodos aplicados para gerar resumos automáticos de texto no contexto da saúde?

O método predominante adotado foi ***Computational Linguistics*** **(CL)**, presente em 17 artigos (73,91%), seguido por ***Mathematical/Statistics***, utilizado em 11 artigos (47,82%), ***Machine Learning*** **(ML)** em 10 artigos (43,47%) e ***Deep Learning*** **(DL)** em 9 artigos (39,13%).

O uso extensivo de *Computational Linguistics* evidencia a continuidade da predominância de métodos tradicionais nessa área. Observa-se também o uso majoritário de métodos híbridos, presentes em 17 artigos (73,91%). Por outro lado, 6 estudos (26,08%) utilizaram apenas um único método, seja *Computational Linguistics* ou *Deep Learning*. Todos os trabalhos diferem entre si, empregando técnicas distintas para a geração dos resumos; a Tabela 11, apresentada anteriormente, descreve as técnicas adotadas pelos métodos propostos. A Figura 6 apresenta a porcentagem de adoção dos métodos.

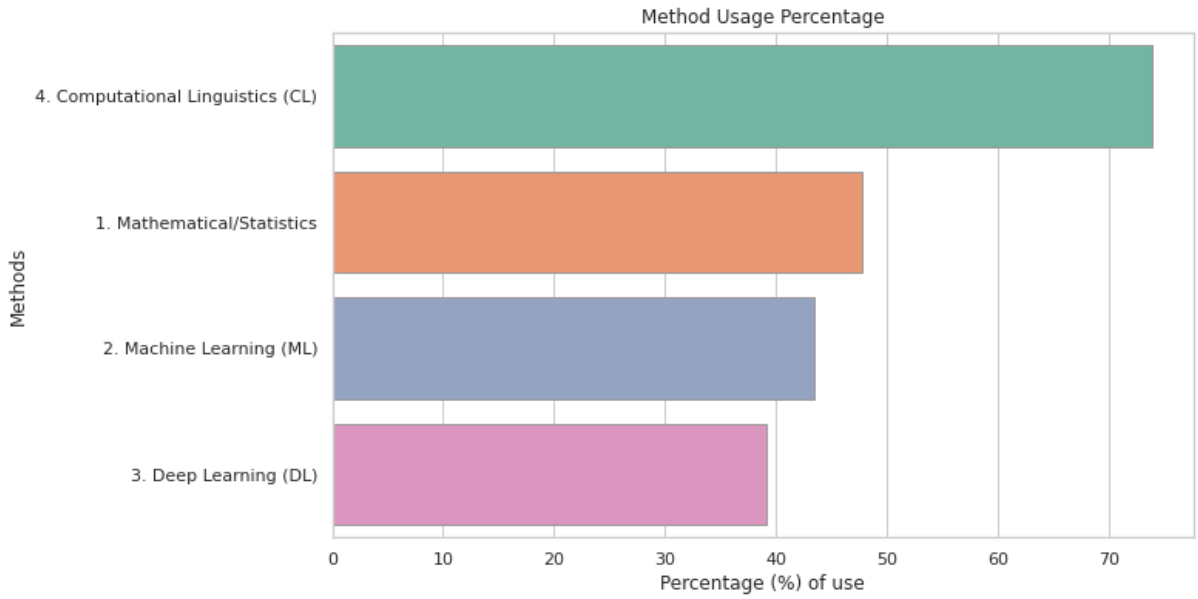


Figura 6 – Percentual de adoção dos métodos.

3.3.2 RQ2 - Em quais áreas do domínio da saúde as técnicas de sumarização de texto são aplicadas?

A principal área de aplicação identificada foi *Medicine*, presente em 9 artigos (39,13%) analisados. Em seguida, destacou-se a área de *Biomedicine*, com 8 artigos (34,78%). Áreas mais específicas, como *Geriatrics* e *Clinical Medicine*, apareceram em 2 artigos cada (8,69%). Diversas outras áreas foram exploradas com menor frequência, tais como *Radiology*, *Internal Medicine*, *Nephrology*, *Neuroscience*, *Neurology*, *Hospital Management*, *Biomedical Informatics*, *Mental Health*, *Psychology*, *Psychiatry* e *Reproductive Medicine*, cada uma representando 1 artigo (4,35%) do total analisado.

Os artigos analisados contemplaram 14 áreas distintas, com uma concentração significativa nas áreas de Medicina e Biomedicine. Essa concentração sugere que essas áreas constituem os principais focos de aplicação das técnicas de sumarização de texto. Entretanto, a presença de subáreas específicas, como *Neurology* e *Mental Health*, indica um interesse crescente na exploração de técnicas computacionais em campos mais especializados. A Tabela 10 descreve as áreas de domínio das aplicações de sumarização de texto por autor.

Tabela 10 – Aplicações de Sumarização de Texto por Domínio da Saúde.

Área	**Autores**
Biomedicine	(KIRMANI et al., 2024; SHANG et al., 2011; WANG et al., 2018; GULDEN et al., 2019; YANG et al., 2024; LEE; UPPAL, 2020; BALAN; GERITS; VANDUFFEL, 2014)
Biomedical Informatics	(AFZAL et al., 2020)
Clinical Medicine	(CAO et al., 2011; OZYEGEN; KABE; CEVIK, 2022; SARKER; MOLLÁ; PARIS, 2016)
Geriatrics	(TANG et al., 2019; MEHRABI et al., 2013)
Hospital Management	(ABULKHAIR et al., 2013)
Medicine	(OZYEGEN; KABE; CEVIK, 2022; YANG et al., 2024; LUO et al., 2023; ADAMS et al., 2021; CABELLO-COLLADO et al., 2024; MORID et al., 2016; AFZAL et al., 2020; LEE; UPPAL, 2020)
Mental Health	(ZIRIKLY et al., 2022)
Nephrology	(SONNTAG; PROFITLICH, 2017)
Neurology/Neuroscience	(BALAN; GERITS; VANDUFFEL, 2014)
Psychology/Psychiatry	(KARYSTIANIS et al., 2018)
Radiology	(CHEN et al., 2024)
Reproductive Medicine	(CAI et al., 2023)

3.3.3 RQ3 - Em quais anos houve o maior número de publicações nesta área?

Com exceção dos anos de 2010, 2012 e 2015, estudos sobre sumarização de texto no domínio da saúde têm sido publicados de forma consistente. Observa-se, de maneira notável, um aumento no número de publicações em 2024, o que sugere um interesse crescente nessa área de pesquisa. A Figura 7 ilustra a distribuição anual das publicações entre os artigos incluídos neste estudo.

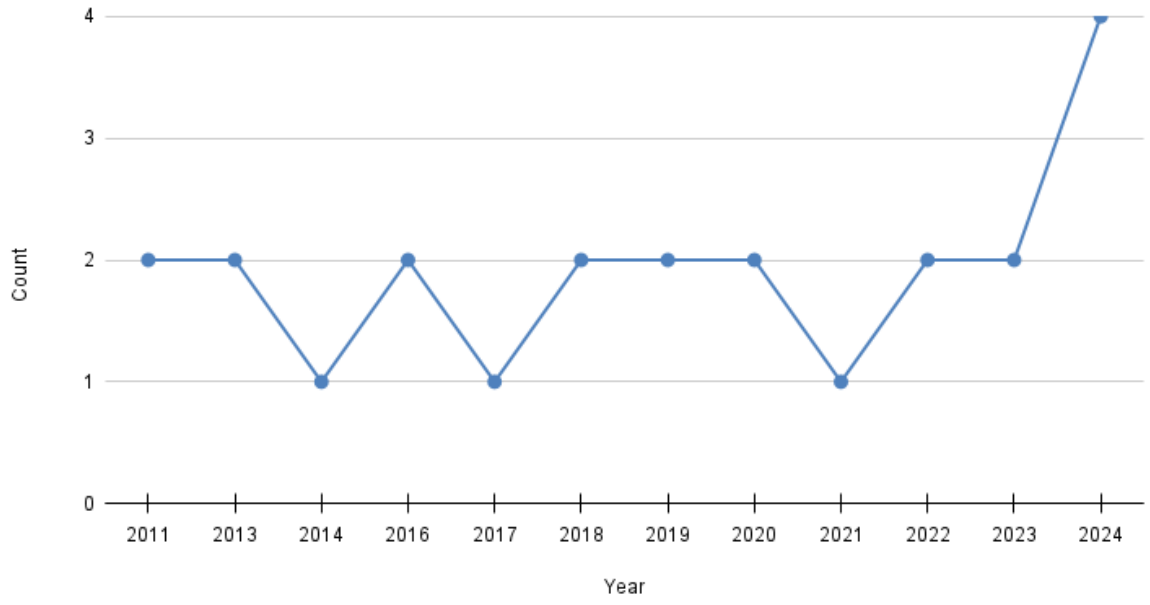


Figura 7 – Artigos por ano de publicação.

3.3.4 RQ4 - Quais países possuem publicações nesta área?

Os trabalhos estão distribuídos entre diversos países. Os USA se destacam com 13 publicações (56,52%), seguidos pela China, com 3 publicações (13,04%), e pela Germany e

Australia, com 2 publicações cada (8,69%). Os demais países, como Belgium, India, Saudi Arabia, Singapore, Spain e o UK, apresentam 1 publicação cada (4,34%). A Figura 8 apresenta o percentual de artigos por país.

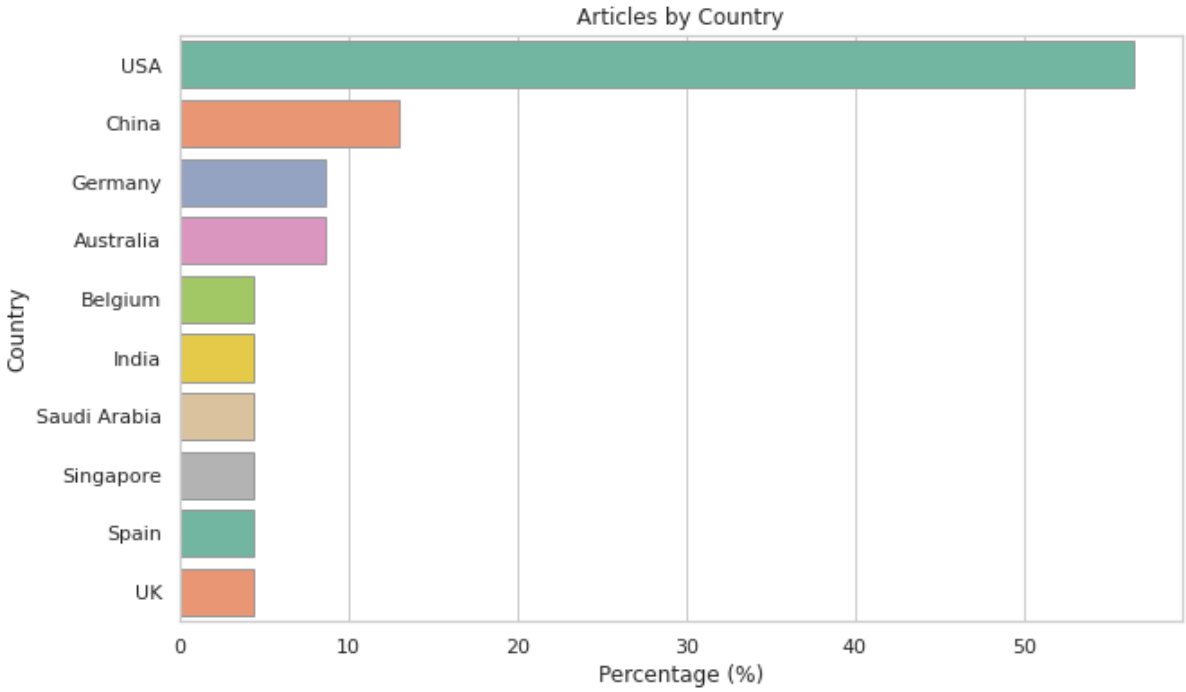


Figura 8 – Percentual de publicações por país.

3.4 Síntese Narrativa e Discussão

Nesta seção, discutimos os principais aspectos e resultados obtidos a partir dos artigos analisados. O objetivo desta pesquisa é investigar os estudos mais recentes sobre sumarização de texto no domínio da saúde, por meio da análise dos métodos desenvolvidos, dos desafios enfrentados, dos benefícios alcançados e das áreas de aplicação.

O volume de textos médicos tem se expandido de forma substancial nas últimas décadas, principalmente em razão da digitalização dos registros em saúde e do crescimento contínuo das publicações científicas no domínio médico. Estimativas atuais sugerem que mais de 3.500 artigos biomédicos são publicados diariamente em diversos periódicos, resultando em um cenário de sobrecarga informacional para clínicos, pesquisadores e profissionais da saúde. O aumento exponencial da quantidade de dados disponíveis, aliado à sua natureza heterogênea — que abrange tanto registros clínicos quanto literatura científica — impõe um desafio significativo para a gestão eficiente da informação.

Nesse contexto, a sumarização automática de texto tem emergido como uma abordagem viável para mitigar os efeitos da sobrecarga informacional. Essas técnicas visam gerar representações condensadas de documentos, preservando seu conteúdo central e as informações essenciais. Ao produzir versões mais curtas e mais facilmente assimiláveis de textos extensos, a sumarização

contribui para aumentar a eficiência na interpretação dos dados e para apoiar processos de tomada de decisão mais rápidos e fundamentados.

Revisões sistemáticas anteriores concentraram-se, predominantemente, na classificação e na descrição sucinta de métodos de sumarização automática aplicados à literatura biomédica e a *Electronic Health Records* (EHRs) (WANG et al., 2021; MISHRA et al., 2014; CHAVES; KESIKU; GARCIA-ZAPIRAIN, 2022), com base nas dimensões propostas por (MISHRA et al., 2014). O estudo apresentado por (HOSSAIN et al., 2023) introduz técnicas empregadas na literatura para a análise de EHR e também destaca e descreve diferentes modelos de *machine learning* (ML) e *Deep Learning* (DL), porém restringe seu escopo a essas abordagens. Esse trabalho também apresenta métricas de avaliação e técnicas de representação textual (por exemplo, word embeddings).

Em contraste, o presente estudo não apenas classifica os métodos de acordo com dimensões específicas, mas também mapeia e descreve as principais abordagens, identifica e discute as principais áreas da saúde contempladas, além de relatar a distribuição das publicações por ano e por país. A revisão não se limita a métodos baseados em ML e DL, incluindo também abordagens tradicionais, que continuam sendo amplamente utilizadas. Ademais, os principais benefícios e desafios dos métodos propostos são agrupados em categorias temáticas e discutidos de forma aprofundada.

3.4.1 Resumos dos Trabalhos

Utilizando técnicas de *Computational Linguistics*, Sonntag e Profitlich (2017) exploraram ferramentas *open-source* para desenvolver uma aplicação integrada que combina um mecanismo de busca e sistemas de visualização, baseada na extração automática de informações a partir de documentos textuais, com o objetivo específico de apoiar a tomada de decisão clínica. Wang et al. (2018) propuseram um novo método de descoberta de meta-padrões para mineração de textos biomédicos, combinando os pontos fortes de abordagens baseadas em cláusulas e em padrões. O principal objetivo foi extrair tanto tipos de relações (meta-padrões sinônimos) quanto instâncias de relações (tuplas) a partir de textos não estruturados, com mínima ou nenhuma supervisão humana. Cai et al. (2023) desenvolveram um sistema de coleta de dados e de suporte à decisão clínica para centros de fertilidade. Por meio do uso de expressões regulares, o RegEMR detecta automaticamente a falência ovariana prematura a partir de Electronic Medical Records (EMRs) em chinês.

Empregando predominantemente técnicas de *Deep Learning*, Luo et al. (2023) projetaram um modelo que integra um template estruturado com conhecimento clínico especializado, *Named Entity Recognition* (NER) e um modelo de classificação de sentenças para orientar a extração de informações a partir de *Electronic Health Records* (EHRs). Além disso, os autores introduziram uma métrica de verificação factual para avaliar os resumos gerados. A abordagem apresentou melhora significativa no desempenho do modelo, produzindo resumos precisos e clinicamente

relevantes. Cabello-Collado et al. (2024) propuseram um sistema automatizado de sumarização para relatórios clínicos, capturando dados provenientes das interações médico-paciente, incluindo fala, entonação e pausas, transcrevendo esses diálogos em texto e utilizando modelos baseados em Transformers para analisar e resumir as informações-chave. O sistema desenvolvido aumentou a confiabilidade e a eficiência da documentação clínica. Chen et al. (2024) desenvolveram um pipeline para o fine-tuning interno de Large Language Models (LLMs) com o objetivo de extrair informações clínicas de laudos radiológicos. Ao realizar o fine-tuning do modelo LLaMA3-8B, os autores demonstraram a viabilidade do uso de modelos *open-source*, alcançando uma pontuação média de F1 de 84,6%, comparável a modelos proprietários de alto custo, como o GPT-4. Yang et al. (2024) introduziram o Ascle, um *toolkit* de NLP projetado para democratizar o acesso a técnicas avançadas de NLP na área da saúde, especialmente para usuários com habilidades limitadas de programação. Entre as funcionalidades do Ascle, destaca-se a sumarização de texto por meio de LLMs, como Pegasus, BigBird, BART, PRIMERA, SciFive e BioBART, sendo o BART o modelo que apresentou o melhor desempenho geral.

Explorando a interseção entre *Deep Learning* e *Computational Linguistics*, Karystianis et al. (2018) investigaram métodos para a extração explícita da gravidade dos sintomas a partir de registros iniciais de avaliação psiquiátrica. Foram implementadas três abordagens: um método baseado em conhecimento, utilizando regras lexicais e padrões sintáticos; um modelo baseado em redes neurais; e um método híbrido. As acurácias obtidas foram de 80,1%, 73,3% e 72,0%, respectivamente, avaliadas por meio da métrics *macro-average MAE*.

Empregando abordagens híbridas que envolvem matemática/estatística e Machine Learning, Tang et al. (2019) aplicaram *Case-Based Reasoning* (CBR) e dados de EHR em tempo real de pacientes idosos em um sistema adaptativo de suporte à decisão clínica, com o objetivo de gerar planos de tratamento aprimorados com base em registros históricos. Um estudo piloto realizado em uma instituição de longa permanência para idosos com doenças crônicas demonstrou redução no tempo de planejamento e aumento na satisfação com os serviços. Ao integrar *Deep Learning* com métodos estatísticos tradicionais e ML, Ozyegen, Kabe e Cevik (2022) desenvolveram técnicas para destacar automaticamente textos em nível de palavra em mensagens médicas voltadas a aplicações de telehealth. Essa abordagem auxiliou os clínicos a identificar rapidamente informações-chave nas mensagens dos pacientes, reduzindo a carga cognitiva e melhorando a eficiência da leitura.

Combinando métodos matemáticos/estatísticos, ML e CL, Cao et al. (2011) desenvolveram o AskHERMES, um sistema de question answering orientado por consultas, destinado a auxiliar clínicos na recuperação de informações relevantes a partir de diversas fontes, incluindo literatura médica e bases de dados online. Para consultas complexas, o sistema superou tanto o Google quanto o UpToDate. (BALAN; GERITS; VANDUFFEL, 2014) conduziram um estudo sobre a aplicação prática de Text Mining (TM) na literatura de reabilitação cognitiva (CR) e aprimoramento cognitivo (CE), especialmente no contexto da estimulação magnética transcraniana (TMS),

demonstrando como o TM pode analisar de forma eficiente grandes volumes de publicações científicas. Sarker, Mollá e Paris (2016) desenvolveram um sistema voltado à medicina baseada em evidências, integrando estatísticas derivadas de corpora, conhecimento de domínio e técnicas avançadas de NLP para mitigar a sobrecarga informacional e gerar resumos concisos e relevantes de pesquisas médicas. Lee e Uppal (2020) propuseram um framework de ranqueamento de sentenças utilizando múltiplos indicadores, random forests e TF-IDF ponderado para sumarização extrativa biomédica. O método superou modelos baseados em um único indicador e demonstrou potencial para integração em sistemas de suporte à decisão clínica, contribuindo para a melhoria da qualidade do serviço e da prestação de cuidados.

Utilizando métodos matemáticos/estatísticos e *Computational Linguistics*, Shang et al. (2011) desenvolveram um sistema para gerar resumos de conceitos biomédicos a partir de múltiplos documentos, com base na extração de relações semânticas, auxiliando pesquisadores no acesso eficiente a informações-chave em meio ao vasto volume da literatura biomédica. Os resultados experimentais superaram o sistema MEAD. Mehrabi et al. (2013) empregaram Conditional Random Fields (CRFs) para extrair eventos causais de artigos do PubMed relacionados ao cuidado geriátrico. Os experimentos, realizados sobre um conjunto de dados anotado manualmente e utilizando uma janela de contexto de atributos antes e após cada evento, alcançaram 84,6% de precisão, 87% de *recall* e *F measure* de 85%, contribuindo para sistemas de suporte à decisão clínica. Abulkhair et al. (2013) propuseram um modelo formativo para um sistema inteligente destinado à automação de relatórios de alta hospitalar (DSRs), com o objetivo de reduzir a carga de trabalho dos médicos e o tempo dedicado à documentação. Gulden et al. (2019) apresentaram um processo para a sumarização automática de descrições de ensaios clínicos por meio de técnicas extrativas. Diversos algoritmos foram avaliados utilizando métricas ROUGE e revisões de completude de conteúdo realizadas por quatro especialistas. Em média, os resumos corresponderam a 25% do tamanho do documento original, sendo o TextRank o método com melhor desempenho. Kirmani et al. (2024) concentraram-se na preservação da semântica de textos biomédicos por meio de modelos semânticos distribucionais (bio-semânticos), explorando word embeddings sob a hipótese distribucional. Essa abordagem não requer análise lexical ou linguística nem depende de informações externas, resultando em melhor desempenho de sumarização.

Morid et al. (2016) exploraram o uso de ML e CL para desenvolver um modelo de classificação baseado em uma rede bayesiana com kernel, utilizando diversos atributos específicos de domínio para extrair sentenças clinicamente úteis de recursos online sintetizados, como o UpToDate.

Ao integrar ML, CL e DL, Afzal et al. (2020) introduziram o Biomed-Summarizer, um framework de sumarização de textos e extração de informações fundamentado na metodologia PICO e em *Deep Learning*. O framework envolve múltiplas etapas, incluindo a identificação de estudos relevantes com base em uma consulta, a seleção de sentenças significativas utilizando a

estrutura PICO e a geração de resumos contextuais de alta qualidade. O método superou modelos tradicionais de ML no apoio à tomada de decisão clínica. Adams et al. (2021) desenvolveram o CLINSUM, um conjunto de dados em larga escala para sumarização multidocumento, contendo 109.000 internações hospitalares, 2 milhões de notas de origem e seus respectivos resumos de "Brief Hospital Course"(BHC). O conjunto inclui notas de “Admission”, “Progress” e “Consultation”, tendo como principal objetivo a geração de resumos concisos e fiéis do curso hospitalar dos pacientes. Por fim, Zirikly et al. (2022) propuseram um framework conceitual voltado à extração de informações em saúde mental para apoiar a avaliação da capacidade funcional, por meio da análise de textos médicos narrativos. Todas as técnicas são detalhadas na Tabela 11.

A análise temporal evidencia que o número de artigos sobre o tema manteve-se relativamente estável ao longo dos anos, com destaque para o ano de 2024, no qual se observa um aumento no interesse pelo uso de métodos mais recentes, como *Deep Learning*, no contexto médico. Esse crescimento reflete a relevância crescente dessas técnicas para a área da saúde.

No que se refere à distribuição geográfica, os resultados indicam que 10 países publicaram estudos sobre o tema, com destaque para os Estados Unidos, responsáveis por 13 artigos (56,52%). A dispersão das publicações entre diferentes continentes constitui um indicativo do interesse global no desenvolvimento de métodos de sumarização de textos aplicados ao contexto médico.

Os trabalhos de Abulkhair et al. (2013), Adams et al. (2021), Afzal et al. (2020), Cabello-Collado et al. (2024), Cai et al. (2023), Kirmani et al. (2024) e Sarker, Mollá e Paris (2016) exploram diferentes métodos de Resumo Automático de Texto aplicados a textos médicos e biomédicos. Por outro lado, Balan, Gerits e Vanduffel (2014), Gulden et al. (2019), Karystianis et al. (2018), Lee e Uppal (2020), Mehrabi et al. (2013), Morid et al. (2016), Shang et al. (2011), Sonntag e Profitlich (2017), Tang et al. (2019), Wang et al. (2018), Yang et al. (2024) e Zirikly et al. (2022) concentraram-se na geração de resumos textuais com extração de informações relevantes de textos médicos e biomédicos, por meio de técnicas de NLP e ML. Por fim, aplicações clínicas como AskHERMES (CAO et al., 2011), BURExtract-Llama (CHEN et al., 2024) e Gsum (CNN) (LUO et al., 2023) foram desenvolvidas para apoiar a tomada de decisão, utilizando técnicas de NLP e ML com o objetivo de aprimorar a eficiência e a precisão das informações médicas.

3.4.2 Dimensões

Com base nas dimensões propostas por Mishra et al. (2014) e Wang et al. (2021), os estudos analisados foram classificados de acordo com o Propósito, o Tipo de Saída, o Tipo de Entrada e a Quantidade de Entrada. No que se refere ao propósito, 56,52% dos estudos implementaram métodos de sumarização genérica, enquanto 43,47% concentraram-se em abordagens orientadas por consulta, gerando resumos personalizados. Destaca-se que apenas 4 (17,39%) dos estudos aplicaram métodos abstrativos, o que evidencia uma lacuna de pesquisa no uso de large language models para a sumarização de textos médicos, sendo a publicação mais antiga datada de 2023. Em

contrapartida, 82% dos estudos empregaram técnicas extrativas. Quanto à quantidade de entrada, 39,13% dos trabalhos processaram resumos de documento único, enquanto 61% abordaram a sumarização multidocumento. Em relação ao tipo de entrada, 43,47% utilizaram *Electronic Health Records* (EHR/EMR), 34,78% concentraram-se na sumarização de textos da literatura, e 4,35% integraram ambas as fontes. Adicionalmente, 17,39% exploraram outros tipos de dados, incluindo ensaios clínicos, registros médicos de idosos, dados biológicos provenientes do MEDLINE e conjuntos de dados biomédicos. A Tabela 11 apresenta uma classificação detalhada das dimensões, métodos e técnicas adotados.

Em aplicações de sumarização orientadas por consulta (query), o foco reside na resolução de situações caso a caso, enfatizando a especificidade do problema. Sistemas guiados por consultas permitem que os usuários recuperem evidências relevantes a partir de grandes volumes de repositórios de dados biomédicos, de modo a apoiar a tomada de decisão clínica complexa (AFZAL et al., 2020). Esses sistemas auxiliam pesquisadores na identificação dos pontos-chave de um determinado tópico (SHANG et al., 2011; TANG et al., 2019), alcançando elevada precisão e aumentando a confiança de clínicos e biólogos (AFZAL et al., 2020; SHANG et al., 2011). Além disso, tais sistemas reduzem o tempo de pesquisa ao concentrar-se nos aspectos mais críticos do conhecimento clínico requerido (AFZAL et al., 2020; SONNTAG; PROFITLICH, 2017; SARKER; MOLLÁ; PARIS, 2016).

Por outro lado, a natureza complexa dos dados biomédicos pode comprometer a qualidade da sumarização orientada por consulta (SARKER; MOLLÁ; PARIS, 2016), e informações cruciais frequentemente encontram-se fora das sentenças diretamente relacionadas à consulta (SHANG et al., 2011). No que diz respeito à disponibilidade de dados, observa-se uma escassez significativa de conjuntos de dados anotados relevantes para dar suporte a aplicações avançadas de NLP no contexto de saúde mental (ZIRIKLY et al., 2022; SARKER; MOLLÁ; PARIS, 2016). Ademais, o conhecimento geral fornecido pelo conjunto central de relações frequentemente carece de descrições específicas em linguagem natural, exigindo a extração de sentenças adicionais de apoio (SHANG et al., 2011).

Aplicações de sumarização genérica, por sua vez, têm como objetivo geral a redução da sobrecarga informacional em grandes volumes de dados (YANG et al., 2024; BALAN; GERITS; VANDUFFEL, 2014; ADAMS et al., 2021), auxiliando na diminuição do tempo necessário para a familiarização com um determinado tema por meio da condensação de sinopses precisas de documentos (GULDEN et al., 2019), aumentando a velocidade, a qualidade e a reprodutibilidade do processamento textual (BALAN; GERITS; VANDUFFEL, 2014), contribuindo para a economia de tempo dos médicos e para a minimização da carga de trabalho (ABULKHAIR et al., 2013; CABELLO-COLLADO et al., 2024), além de apoiar a tomada de decisão clínica (CAI et al., 2023; KARYSTIANIS et al., 2018; MEHRABI et al., 2013).

Consequentemente, o desenvolvimento de resumos genéricos apresenta maior complexidade. Esses resumos podem sofrer com redundância de informações (ADAMS et al., 2021),

e a complexidade linguística inerente ao domínio da saúde também representa um desafio significativo (ADAMS et al., 2021). Quando se empregam técnicas de machine learning, torna-se necessária a anotação humana dos dados de treinamento, um processo dispendioso em termos de tempo e esforço (WANG et al., 2018). Ademais, tais dados são extremamente sensíveis e onerosos (LUO et al., 2023), ou podem até mesmo não estar disponíveis (GULDEN et al., 2019). O uso de redes neurais demanda grandes volumes de dados e recursos computacionais significativos (KIRMANI et al., 2024), enquanto a natureza específica dos casos nos dados biomédicos limita a capacidade de generalização dos modelos (CAI et al., 2023).

Tabela 11 – Critérios de classificação dos estudos, suas dimensões, métodos e técnicas. Ordenados por ano de publicação.

Authors	Year	Methods	Techniques	Input Quant.	Input Type	Purpose	Output Type
(CAO et al., 2011)	2011	Math/Stats, ML, CL	SVM, Conditional Random Field (CRF), BM25, Hierarquical Clustering, Unified Medical Language System (UMLS)	Multi	Literature	Query	Extractive
(SHANG et al., 2011)	2011	Math/Stats, CL	SemRep, Information Retrieval, Okapi BM25, Unified Medical Language System (UMLS)	Multi	Others	Query	Extractive
(ABULKHAIR et al., 2013)	2013	Math/Stats, CL	Greed Algorithm	Multi	EHR/EMR	Generic	Extractive
(MEHRABI et al., 2013)	2013	Math/Stats, CL	Conditional Random Fields (CRF), Part-of-Speech (POS) Tag, Shallow Parser	Single	Literature	Generic	Extractive
(BALAN; GERITS; VANDUFFEL, 2014)	2014	Math/Stats, ML, CL	Mallet, Text to Matrix Generator (TMG), KH Coder, Anote2, BioRAT	Multi	Literature	Generic	Extractive
(MORID et al., 2016)	2016	ML, CL	Kernel Based Bayesian Newtork	Single	Literature	Query	Extractive
(SARKER; MOLLÁ; PARIS, 2016)	2016	Math/Stats, ML, CL	Maximum Marginal Relevance (MMR), MetaMap, PIBOSO, Unified Medical Language System (UMLS)	Single	Literature	Query	Extractive
(SONNTAG; PROFITLICH, 2017)	2017	CL	UIMA, Averbis, Shallow Text Parsing	Multi	EHR/EMR	Query	Extractive
(KARYSTIANIS et al., 2018)	2018	DL, CL	Lexical Rule Based Approach, Neural Network, Hybrid Method,	Single	EHR/EMR	Generic	Extractive
(WANG et al., 2018)	2018	CL	CPIE, Meta-Pattern Discovery, NER, Clause Extraction	Multi	Literature	Generic	Extractive
(GULDEN et al., 2019)	2019	Math/Stats, CL	LexRank, TextRank, LSA, Luhn, SumBasic, KLSum	Single	Others	Generic	Extractive
(TANG et al., 2019)	2019	Math/Stats, ML	TF-IDF, Case Base Reasoning (CBR), Naive Bayes	Single	Others	Query	Extractive
(AFZAL et al., 2020)	2020	ML, DL, CL	PICO, Bi-LSTM, JS2E	Multi	Literature	Query	Extractive
(LEE; UPPAL, 2020)	2020	Math/Stats, ML, CL	MINTS, Indicative Summarization	Single	Literature	Generic	Extractive
(ADAMS et al., 2021)	2021	ML, DL, CL	Linked Entity Extraction, Local Coherence, Unified Medical Language System (UMLS)	Multi	EHR/EMR	Generic	Extractive
(OZYEGEN; KABE; CEVIK, 2022)	2022	Math/Stats, ML, DL	TF-IDF, Word2Vec, LIME, Dense NN, CNN-CRF, LSTM, BERT	Single	EHR/EMR	Query	Extractive
(ZIRIKLY et al., 2022)	2022	ML, DL, CL	i2b2 Annotation, Logistic Regression, Med-BERT embedding, SVM	Single	EHR/EMR	Query	Extractive
(CAI et al., 2023)	2023	CL	REGEX, Rule Based Summarizer	Single	EHR/EMR	Generic	Extractive
(LUO et al., 2023)	2023	DL	BART, *Named Entity Recognition* (NER), In-context learning (ICL) with Summary Template	Single	EHR/EMR	Generic	Abstractive
(CABELLO-COLLADO et al., 2024)	2024	DL	Automatic speech recognition (ASR), BART, DISTILLBART, BART-SAMSUM, GPT2, BERT2BERT, T5	Single	EHR/EMR	Generic	Abstractive
(CHEN et al., 2024)	2024	DL	Llama3-8b, GPT-4, Fine-Tuning	Single	EHR/EMR	Query	Abstractive
(KIRMANI et al., 2024)	2024	Math/Stats, CL	Bio-Semantic Models, BioBERT, Word2Vec, K-Means, Score Based Ranking	Single	Others	Generic	Extractive
(YANG et al., 2024)	2024	DL	RAG, Ranking, TextRank, Bart, Pegasus, PRIMERA, Biobart, BigBirdPegasus, Unified Medical Language System (UMLS)	Multi	Literature and EHR/MDR	Generic	Abstractive

3.4.3 Principais Benefícios

Os autores destacaram diversos benefícios alcançados por meio das metodologias propostas, incluindo: otimização/melhoria da busca por informações (65,22%), melhoria no processo de tomada de decisão (43,48%), melhoria de processos operacionais (39,13%), melhoria da qualidade dos resumos (30,43%), resultados comparáveis ou superiores ao State of the Art (SOTA) (30,43%), redução de tempo (26,09%), redução da sobrecarga de informação (17,39%), criação de um novo conjunto de dados (17,39%), redução da carga cognitiva (13,04%), redução de custos (8,70%), democratização do acesso à informação (4,35%), melhoria de processos operacionais (4,35%), melhoria/apoio à tomada de decisão (4,35%), melhoria da legibilidade dos resumos (4,35%) e replicabilidade (4,35%). A Tabela 12 detalha os benefícios citados por cada autor.

Tabela 12 – Benefícios citados por autor.

Benefícios	**Autores**
Otimização/melhoria da busca por informações	(SHANG et al., 2011), (TANG et al., 2019), (OZYEGEN; KABE; CEVIK, 2022), (CABELLO-COLLADO et al., 2024), (BALAN; GERITS; VANDUFFEL, 2014), (SARKER; MOLLÁ; PARIS, 2016), (AFZAL et al., 2020), (WANG et al., 2018), (GULDEN et al., 2019), (CHEN et al., 2024), (YANG et al., 2024), (CAI et al., 2023), (MORID et al., 2016), (MEHRABI et al., 2013), (CAO et al., 2011)
Melhoria/apoio à tomada de decisão	(LEE; UPPAL, 2020), (ADAMS et al., 2021), (CABELLO-COLLADO et al., 2024), (LUO et al., 2023), (KARYSTIANIS et al., 2018), (CAI et al., 2023), (ZIRIKLY et al., 2022), (SONNTAG; PROFITLICH, 2017), (MORID et al., 2016), (AFZAL et al., 2020)
Melhoria de processos operacionais	(LEE; UPPAL, 2020), (ADAMS et al., 2021), (OZYEGEN; KABE; CEVIK, 2022), (CABELLO-COLLADO et al., 2024), (SARKER; MOLLÁ; PARIS, 2016), (KIRMANI et al., 2024), (YANG et al., 2024), (SONNTAG; PROFITLICH, 2017), (CAO et al., 2011)
Melhoria da qualidade dos resumos	(SHANG et al., 2011), (ABULKHAIR et al., 2013), (LUO et al., 2023), (KIRMANI et al., 2024), (SARKER; MOLLÁ; PARIS, 2016), (YANG et al., 2024), (AFZAL et al., 2020)
Resultados comparáveis ou superiores ao State of the Art (SOTA)	(SHANG et al., 2011), (ABULKHAIR et al., 2013), (KIRMANI et al., 2024), (WANG et al., 2018), (CHEN et al., 2024), (CAI et al., 2023), (MORID et al., 2016)
Redução de tempo	(ABULKHAIR et al., 2013), (TANG et al., 2019), (OZYEGEN; KABE; CEVIK, 2022), (YANG et al., 2024), (SONNTAG; PROFITLICH, 2017), (CAO et al., 2011)
Redução da sobrecarga de informação	(LEE; UPPAL, 2020), (SARKER; MOLLÁ; PARIS, 2016), (ABULKHAIR et al., 2013), (YANG et al., 2024)
Criação de um novo conjunto de dados	(GULDEN et al., 2019), (KARYSTIANIS et al., 2018), (AFZAL et al., 2020), (CABELLO-COLLADO et al., 2024)
Redução da carga cognitiva	(ABULKHAIR et al., 2013), (OZYEGEN; KABE; CEVIK, 2022), (YANG et al., 2024)
Redução de custos	(CHEN et al., 2024), (OZYEGEN; KABE; CEVIK, 2022)
Democratização do acesso à informação	(YANG et al., 2024)
Melhoria de processos	(TANG et al., 2019)
Melhoria/apoio à tomada de decisão	(MEHRABI et al., 2013)
Melhoria da legibilidade dos resumos	(SHANG et al., 2011)
Melhoria/apoio à tomada de decisão	(YANG et al., 2024)
Replicabilidade	(KIRMANI et al., 2024)

3.4.4 Principais Desafios

Por sua vez, os desafios identificados na literatura revisada mostraram-se variados, com destaque para as limitações nos métodos propostos, mencionadas em 20 (86,96%) artigos. Em seguida, a complexidade linguística foi reportada em 18 (78,26%) artigos, enquanto a escassez e as limitações de dados apareceram em 14 (60,87%) artigos, e as limitações das métricas de avaliação e da qualidade dos dados foram mencionadas em 10 (43,48%) artigos cada. A dependência de especialistas foi citada em 5 (21,74%) artigos, e os custos computacionais e de processamento foram apontados como desafios em 4 (17,39%) artigos. A Tabela 13 detalha os desafios enfrentados por cada autor.

Tabela 13 – Desafios citados por autor.

Desafios	**Autores**
Custos computacionais e de processamento	(KIRMANI et al., 2024), (YANG et al., 2024), (ABULKHAIR et al., 2013), (SONNTAG; PROFITLICH, 2017)
Qualidade dos dados	(CAO et al., 2011), (TANG et al., 2019), (SHANG et al., 2011), (GULDEN et al., 2019), (ADAMS et al., 2021), (CABELLO-COLLADO et al., 2024), (AFZAL et al., 2020), (LEE; UPPAL, 2020), (SONNTAG; PROFITLICH, 2017), (CAI et al., 2023)
Escassez e limitações de dados	(CAO et al., 2011), (OZYEGEN; KABE; CEVIK, 2022), (KIRMANI et al., 2024), (WANG et al., 2018), (YANG et al., 2024), (SARKER; MOLLÁ; PARIS, 2016), (MEHRABI et al., 2013), (LUO et al., 2023), (CABELLO-COLLADO et al., 2024), (LEE; UPPAL, 2020), (ZIRIKLY et al., 2022), (KARYSTIANIS et al., 2018), (CHEN et al., 2024), (CAI et al., 2023)
Dependência de especialistas	(CAO et al., 2011), (TANG et al., 2019), (KIRMANI et al., 2024), (SARKER; MOLLÁ; PARIS, 2016), (BALAN; GERITS; VANDUFFEL, 2014)
Limitações no método proposto	(CAO et al., 2011), (OZYEGEN; KABE; CEVIK, 2022), (TANG et al., 2019), (SHANG et al., 2011), (WANG et al., 2018), (GULDEN et al., 2019), (YANG et al., 2024), (SARKER; MOLLÁ; PARIS, 2016), (MEHRABI et al., 2013), (ABULKHAIR et al., 2013), (LUO et al., 2023), (CABELLO-COLLADO et al., 2024), (MORID et al., 2016), (LEE; UPPAL, 2020), (ZIRIKLY et al., 2022), (SONNTAG; PROFITLICH, 2017), (BALAN; GERITS; VANDUFFEL, 2014), (KARYSTIANIS et al., 2018), (CHEN et al., 2024), (CAI et al., 2023)
Limitações das métricas de avaliação	(OZYEGEN; KABE; CEVIK, 2022), (KIRMANI et al., 2024), (WANG et al., 2018), (GULDEN et al., 2019), (YANG et al., 2024), (SARKER; MOLLÁ; PARIS, 2016), (LUO et al., 2023), (AFZAL et al., 2020), (BALAN; GERITS; VANDUFFEL, 2014), (CHEN et al., 2024)
Complexidade linguística	(CAO et al., 2011), (OZYEGEN; KABE; CEVIK, 2022), (KIRMANI et al., 2024), (WANG et al., 2018), (YANG et al., 2024), (SARKER; MOLLÁ; PARIS, 2016), (MEHRABI et al., 2013), (LUO et al., 2023), (ADAMS et al., 2021), (CABELLO-COLLADO et al., 2024), (MORID et al., 2016), (AFZAL et al., 2020), (LEE; UPPAL, 2020), (ZIRIKLY et al., 2022), (SONNTAG; PROFITLICH, 2017), (BALAN; GERITS; VANDUFFEL, 2014), (KARYSTIANIS et al., 2018), (CHEN et al., 2024)

3.5 Considerações Finais

Este trabalho realizou um mapeamento sistemático com o objetivo de identificar e caracterizar o conjunto de pesquisas primárias que abordam o uso de métodos de sumarização de textos aplicados ao domínio da saúde. Dos 447 trabalhos recuperados nas bases de dados científicas, 23 (5,15%) foram aceitos com base nos critérios de inclusão e exclusão. Destaca-se a aplicação combinada de diferentes metodologias e, entre os principais métodos empregados, a Linguística Computacional sobressaiu-se, estando presente em 17 (73,91%) dos artigos, seguida por métodos Matemáticos/Estatísticos, utilizados em 11 (47,82%), *machine learning* (ML) em 10 (43,47%) e *Deep Learning* (DL) em 9 (39,13%) artigos. Esse resultado evidencia a continuidade da relevância dos métodos tradicionais de sumarização de textos. Quanto às áreas de estudo, observa-se uma ampla variedade de domínios, demonstrando a flexibilidade e a aplicabilidade dos métodos em diferentes contextos. A maioria dos trabalhos aborda o problema de forma orientada a consultas e considerando documentos únicos, o que evidencia a limitação de escalabilidade para o processamento de grandes volumes de texto. Ademais, a maior parte dos estudos trata o problema por meio de abordagens extrativas, indicando uma lacuna de pesquisa no uso de métodos abstrativos.

Por fim, acredita-se que este trabalho forneça resultados relevantes para a comunidade acadêmica, oferecendo uma fonte de referência sobre os principais métodos utilizados em diferentes áreas para a resolução de problemas por meio da sumarização de textos, podendo auxiliar pesquisadores e instituições de saúde.

A seguir, o próximo capítulo discute os trabalhos relacionados no campo de resumo automático extrativo de texto por meio de algoritmos bioinpsirados, que fundamentam os métodos propostos neste trabalho para endereçar os desafios reportados nos estudos mapeados, tais como

os custos computacionais e de processamento, a dependência de especialistas de domínio e a complexidade linguística.

4

Algoritmos Bioinspirados Aplicados em Sumarização

Este Capítulo apresenta os trabalhos relacionados à algoritmos bioinspirados com foco em resumo automático de texto, que serviram de base para os métodos propostos nos Capítulos 5 e 6.

4.1 Revisão da Literatura

Na literatura sobre sumarização automática de textos extrativa utilizando métodos de otimização bioinspirados, diversas abordagens têm sido propostas, tais como *Differential Evolution* (DE) (ALIGULIYEV, 2009; ALGULIEV; ALIGULIYEV; MEHDIYEV, 2011; ALGULIEV; ALIGULIYEV; ISAZADE, 2012; ALGULIYEV; ALIGULIYEV; ISAZADE, 2015; SAINI; SAHA; BHATTACHARYYA, 2019; SAINI et al., 2019), *Genetic Algorithms* (KUMAR et al., 2014; ABBASI-GHALEHTAKI; KHOTANLOU; ESMAEILPOUR, 2016; AL-RADAIDEH; BATAINEH, 2018; ALQAISI; GHANEM; QAROUSH, 2020; MOJRIAN; MIRROSHANDEL, 2021), métodos baseados em *Swarm Intelligence* (SANCHEZ-GOMEZ; VEGA-RODRíGUEZ; PéREZ, 2018; SANCHEZ-GOMEZ; VEGA-RODRíGUEZ; PéREZ, 2020; TOMER; KUMAR, 2022), abordagens baseadas em algoritmos meméticos (MENDOZA et al., 2014; SANCHEZ-GOMEZ; VEGA-RODRíGUEZ; PéREZ, 2022) e métodos com foco na formulação de problemas multiobjetivo (HUANG et al., 2010; SANCHEZ-GOMEZ; VEGA-RODRíGUEZ; PéREZ, 2024). Outras abordagens incluem *Integer Linear Programming* (ILP) (VERMA; VERMA; PAL, 2022) e métodos *Evolutionary Semantic-Based* (HERNANDEZ-CASTANEDA et al., 2020). A Tabela 14 apresenta um resumo das abordagens previamente propostas.

Tabela 14 – Trabalhos relacionados à sumarização automática de textos extrativa baseada em métodos bioinspirados:

Ano	Autor	Método/Classe	Descrição
2009	Aliguliyev (2009)	*Differential Evolution* (DE)	Formulação multiobjetivo; método genérico de sumarização baseado em agrupamento de sentenças utilizando DE.
2010	Huang et al. (2010)	*Multi-objective* (4 funções)	Sumarização orientada a consulta com objetivos de cobertura, significância, redundância e coerência.
2011	Alguliev, Aliguliyev e Mehdiyev (2011)	*Adaptive* DE	Modelo de sumarização multidocumento que extrai sentenças-chave e reduz redundância por meio de DE adaptativo.
2012	Alguliev, Aliguliyev e Isazade (2012)	DESAMC (*self-Adaptive* DE)	Problema modelado como *p-median*; DE com parâmetros de mutação e cruzamento autoajustáveis.
2014	Kumar et al. (2014)	*Genetic* (*Genetic*-CBR)	Sumarização multidocumento com detecção de relações entre documentos; pontuação de sentenças via raciocínio *Fuzzy*.
2014	Mendoza et al. (2014)	*Memetic* (MA-SingleDocSum)	Sumarização de documento único combinando operadores genéticos e busca local guiada.
2015	Alguliyev, Aliguliyev e Isazade (2015)	*Boolean programming* + DE	Abordagem não supervisionada para sumarização de documentos únicos e múltiplos, otimizando relevância, redundância e comprimento.
2016	Abbasi-ghalehtaki, Khotanlou e Esmaeilpour (2016)	*Genetic* + PSO + *Fuzzy*	Modelo FPGAC que otimiza pesos de características com PSO e GA; sistema de lógica *Fuzzy* para pontuação final das sentenças.
2018	Al-Radaideh e Bataineh (2018)	*Hybrid Genetic* (ASDKGA)	Sumarização de documentos políticos em árabe; combinação de conhecimento de domínio e características estatísticas com GA.
2018	Sanchez-Gomez, Vega-Rodríguez e Pérez (2018)	MOABC (*Artificial Bee Colony*)	Sumarização multiobjetivo considerando comprimento, cobertura de conteúdo e redundância.
2019	Saini, Saha e Bhattacharyya (2019)	MOBDE (*binary Multi-objective* DE)	Sumarização de microblogs (MOO*Tweets*umm); otimização simultânea de comprimento e antirredundância; exploração de operadores genéticos como SOM.
2019	Saini et al. (2019)	MODE, MGWO, MWCA	Integração de SOM com algoritmos multiobjetivo (DE, *Grey Wolf*, *Water Cycle*) para seleção de sentenças representativas a partir de *clusters*.
2020	Sanchez-Gomez, Vega-Rodríguez e Pérez (2020)	MOABC/D	Versão baseada em decomposição com implementação paralela assíncrona em arquiteturas *multicore*.
2020	Alqaisi, Ghanem e Qaroush (2020)	NSGA-II	Otimização multiobjetivo global para maximizar cobertura, diversidade e relevância.
2020	Hernandez-Castaneda et al. (2020)	*Evolutionary Semantic-Based*	Aumenta a cobertura por meio de agrupamento semântico e melhora a precisão via detecção de palavras-chave.
2021	Mojrian e Mirroshandel (2021)	*Quantum-Inspired* GA (QIGA)	Sumarização genérica multidocumento; otimiza combinação linear de cobertura, relevância e redundância com rotação quântica adaptativa.
2022	Verma, Verma e Pal (2022)	ILP + *Teaching-Learning* + *Fuzzy*	Modelo híbrido em dois estágios (agrupamento + otimização); combina distâncias semânticas com inferência *Fuzzy* para pontuação das sentenças.
2022	Tomer e Kumar (2022)	*Firefly-based* (FbTS)	Sumarização multidocumento utilizando uma função de aptidão única (relação documento-tópico, coesão e legibilidade).
2022	Sanchez-Gomez, Vega-Rodríguez e Pérez (2022)	MOSFLA (*Shuffled Frog-Leaping*)	Sumarização multiobjetivo orientada a consulta avaliada no conjunto de dados TAC; obteve melhorias significativas em ROUGE.
2024	Sanchez-Gomez, Vega-Rodríguez e Pérez (2024)	IMOVNS (*Variable Neighborhood Search*)	Algoritmo orientado a consulta baseado em busca em vizinhança variável multiobjetivo orientada por indicadores.

Todas as abordagens revisadas empregaram métricas ROUGE em seus experimentos, principalmente ROUGE-1 e ROUGE-2. No entanto, alguns estudos também incluíram outras variações do ROUGE, como ROUGE-S, ROUGE-SU, ROUGE-SU4 e ROUGE-L, em suas avaliações de desempenho. Ademais, a maioria dos estudos utilizou alguma variação do conjunto de dados da *Document Understanding Conference* (DUC), sendo o DUC2002 o mais frequentemente empregado, seguido pelo DUC2001. Dessa forma, neste estudo e em seus experimentos, as métricas principais ROUGE-1 e ROUGE-2, juntamente com os conjuntos de dados DUC2002 e DUC2001, foram adotados para fins comparativos.

Na literatura científica sobre métodos de otimização bioinspirados para sumarização automática de textos extrativa, diversas limitações podem ser identificadas. Entre elas, destacam-se limitações relacionadas à avaliação, à complexidade dos métodos, à dependência de especialistas humanos, ao elevado uso de recursos computacionais, aos desafios de ajuste de parâmetros e às limitações metodológicas.

Limitações na Avaliação: diversos estudos apresentam restrições metodológicas em seus processos de avaliação. em Aliguliyev (2009), a medida de dissimilaridade empregada baseia-se na Distância Normalizada do Google (NGD), a qual, por não satisfazer a desigualdade triangular, configura-se apenas como uma medida de dissimilaridade, e não como uma métrica de distância. em Alguliev, Aliguliyev e Isazade (2012), o método extrativo genérico DESAMC+DocSum foi avaliado utilizando os datasets DUC2005 e DUC2006, originalmente projetados para sumarização orientada a consultas (query-based), o que também ocorre em Mojrian e Mirroshandel (2021), onde um sumarizador genérico é testado em corpora destinados à sumarização query-focused, sem considerar a consulta fornecida. em Mendoza et al. (2014), os autores não aplicaram testes de hipótese não paramétricos para validação estatística dos resultados, reconhecendo essa limitação como uma perspectiva de trabalhos futuros. em Al-Radaideh e Bataineh (2018), apesar da avaliação em 200 documentos do corpus KALIMAT, o teste adicional no corpus EASC foi restrito a apenas 21 documentos, comprometendo a robustez dos resultados. De forma semelhante, Saini, Saha e Bhattacharyya (2019) evidenciam que a eficácia dos operadores baseados em SOM varia conforme o dataset e o problema analisado, indicando a ausência de generalização do método. Por fim, em Alqaisi, Ghanem e Qaroush (2020), o uso do dataset DUC2002 traduzido via Google Translator e posteriormente validado manualmente, introduzindo potenciais imprecisões decorrentes da dependência de tradução automática e intervenção humana.

Complexidade do método: o método proposto por Abbasi-ghalehtaki, Khotanlou e Esmaeilpour (2016) é uma combinação de vários algoritmos (*Fuzzy* Logic, PSO-GA e CLA-ABC). Essa alta complexidade implica uma dependência de ajuste e otimização cuidadosa de múltiplos conjuntos de parâmetros para cada um dos algoritmos evolutivos e de swarm utilizados. Já em Verma, Verma e Pal (2022), a complexidade de tempo total do algoritmo proposto não pode ser decidida devido à complexidade não determinada do Algoritmo de Otimização Baseada em Ensino-Aprendizagem (TLBO) e à dificuldade de implementação do sistema de controle *Fuzzy*

em laços de controle. em Sanchez-Gomez, Vega-Rodríguez e Pérez (2024), como um algoritmo de Otimização Multi-Objetivo (MOO), ele resulta em um conjunto de soluções não-dominadas (*Pareto set*), exigindo um método adicional (como o método de consenso, que forneceu os melhores resultados) para selecionar uma única solução final.

Dependência humana/experts: em Kumar et al. (2014), o modelo de raciocínio *Fuzzy* depende criticamente de 81 regras IF-THEN que foram construídas por especialistas de domínio. Já em Abbasi-ghalehtaki, Khotanlou e Esmaeilpour (2016), a pontuação final das sentenças depende criticamente das 24 regras definidas por especialistas, e seu sistema de lógica *Fuzzy* utilizou cerca de 150 regras IF-THEN, que foram definidas por três especialistas humanos. No mesmo sentido, em Verma, Verma e Pal (2022), existe uma dependência crítica de Conhecimento Humano, pois o sistema de inferência *Fuzzy* requer um conjunto de regras IF-THEN geradas por humanos. Foi utilizada uma base de conhecimento total de 205 regras definidas por três avaliadores. em Al-Radaideh e Bataineh (2018), a abordagem requer a preparação manual de conhecimento de domínio, que envolve a identificação, filtragem e categorização de uma lista de 500 palavras-chave políticas por um especialista político. A escalabilidade e a adaptabilidade a novos domínios dependem da capacidade de construir e manter essas bases de conhecimento. em Saini, Saha e Bhattacharyya (2019), o sistema utiliza o WMD, que depende de um modelo word2vec pré-treinado em um corpus de 53 milhões de *Tweets* de desastres. Essa dependência de regras manuais pode limitar a generalização e a adaptação do modelo a novos domínios.

Limitação no Método: a abordagem em Huang et al. (2010), Sanchez-Gomez, Vega-Rodríguez e Pérez (2024) foi desenvolvida, implementada e avaliada para extração genérica baseada em consulta (query-based). Enquanto que em outros trabalhos, foram desenvolvidos, implementados e avaliados métodos para extração genérica *single-document*, não *multi-document* (ALIGULIYEV, 2009; MENDOZA et al., 2014; ABBASI-GHALEHTAKI; KHOTANLOU; ESMAEILPOUR, 2016; AL-RADAIDEH; BATAINEH, 2018; SAINI et al., 2019). Os autores utilizaram uma base de dados limitada, contendo somente notícias de desastres naturais (KUMAR et al., 2014), documentos políticos em árabe (AL-RADAIDEH; BATAINEH, 2018) e *Tweets* (SAINI; SAHA; BHATTACHARYYA, 2019). E o método desenvolvido se limitou ao idioma árabe (AL-RADAIDEH; BATAINEH, 2018; ALQAISI; GHANEM; QAROUSH, 2020).

Uso Intensivo em Recursos Computacionais: em Huang et al. (2010), as técnicas exatas de busca pela solução ótima propostas para o sumário são frequentemente inaplicáveis devido à sua alta complexidade computacional; Para alcançar uma velocidade de busca razoável, o método se limita a selecionar apenas as 100 sentenças mais bem ranqueadas em termos de cobertura e significância como soluções candidatas. Já (SAINI; SAHA; BHATTACHARYYA, 2019), embora a abordagem "without SOM"seja mais eficiente, a aplicação em tempo real de abordagens evolutivas ainda pode ser desafiadora; Os métodos de *ensembling* são evitados por serem demorados em cenários em tempo real. em Saini et al. (2019), o ESDS_SMODE leva mais tempo de CPU para ser executado do que os outros dois métodos (ESDS_MGWO e

ESDS_MWCA). A média de tempo de CPU para o DUC2001 é de 78.99 segundos/documento, e para o DUC2002 é de 50.34 segundos/documento. em Verma, Verma e Pal (2022), o método exato (B&B) não é adequado para grandes volumes de dados porque exige muito tempo computacional; O tempo de execução da Solução Exata (B&B) é significativamente maior do que o método de aproximação (FEKM_Gap*), tornando-o inviável para documentos longos. Para um documento com 11 sentenças e sumário de 5 sentenças, o B&B levou 1183 segundos, enquanto o FEKM levou 0.937 segundos.

Todas as abordagens revisadas empregaram métricas ROUGE em seus experimentos, principalmente ROUGE-1 e ROUGE-2. No entanto, alguns estudos também incluíram outras variações do ROUGE, como ROUGE-S, ROUGE-SU, ROUGE-SU4 e ROUGE-L, em suas avaliações de desempenho. Além disso, a maioria dos estudos utilizou alguma variação do conjunto de dados da *Document Understanding Conference* (DUC), sendo o DUC2002 o mais frequentemente empregado, seguido pelo DUC2001. Dessa forma, neste estudo e em seus experimentos, foram adotadas, para fins comparativos, as métricas ROUGE-1 e ROUGE-2, juntamente com os conjuntos de dados DUC2002 e DUC2001.

Tais limitações foram endereçadas nos métodos propostos *Maximum Relevance with Minimum Redundancy using Shuffle Frog-Leaping Algorithm* (MRMRSFLA) e *Holistic Text Summarization with the Shuffled Frog-Leaping Algorithm* (HSSFLA) nos Capítulos 5 e 6, pois eles foram avaliados de forma robusta, definindo os parâmetros no DUC2001 e mensurando seu desempenho também no DUC2002, mostrando sua capacidade de generalização. Ambos são uma extensão do *Shuffle Frog Leaping Algorithm*, um algoritmo memético de baixa complexidade, buscando testar duas hipóteses diferentes e opostas sobre resumos automáticos: resumo baseado em tópicos e holístico. Os métodos propostos não dependem de experts em nenhuma etapa do *pipeline*, são não supervisionados, foram formulados para serem genéricos e são aplicáveis para ambos métodos *single* ou *multi-document*, e independende do domínio e linguagem (idioma). Os métodos também não desconsideram soluções candidatas que excedem o limite máximo pré-definido, mas lida com elas penalizando conforme o tamanho excedido, para que boas soluções não sejam perdidas caso excedam minimamente o tamanho definido.

4.2 Definição do Problema

Esta seção apresenta a abordagem genérica para a sumarização de textos multi-documento. Na literatura existente, métodos baseados em vetores de palavras estão entre as técnicas mais comumente empregadas. Essa abordagem representa cada sentença como um vetor de palavras e determina a similaridade entre sentenças utilizando critérios específicos, como a similaridade do cosseno. Em seguida, são introduzidas a representação vetorial das sentenças, o critério de similaridade adotado e o critério de informatividade.

4.2.1 Representação de Sentenças

Inicialmente, cada sentença é representada como um vetor de palavras. Seja $T = t_1, t_2, \ldots, t_m$ o conjunto de todos os termos únicos presentes na coleção de documentos D, em que m representa o número total de termos. Cada sentença s_i em D pode, então, ser expressa como um vetor de dimensão m, formulado como $s_i = (w_{i1}, w_{i2}, \ldots, w_{im})$, em que $i = 1, 2, \ldots, n$, e n denota o número total de sentenças. Nessa representação, cada componente corresponde ao peso do termo t_k na sentença s_i. O peso w_{ik} é calculado por meio do método *term-frequency inverse-sentence-frequency* $tf - isf$, conforme descrito em Salton e Buckley (1988), e é definido pela Equação (4.1):

$$w_{ik} = tf_{ik} \cdot \log(n/n_k) \tag{4.1}$$

Em que tf_{ik} representa a frequência do termo t_k na sentença s_i, n representa o número total de sentenças no corpus ou coleção (D), e n_k denota o número de sentenças em D que contêm o termo t_k.

4.2.2 Medida de Similaridade do Cosseno

A medida de similaridade do cosseno é definida com base na representação das sentenças previamente introduzida. Essa métrica de similaridade avalia o grau de semelhança entre duas sentenças, s_i e s_j, na coleção de documentos D. Ela é calculada por meio da Equação 4.2:

$$cosim(s_i, s_j) = \frac{\sum_{k=1}^{m} w_{ik} w_{jk}}{\sqrt{\sum_{k=1}^{m} w_{ik}^2} \cdot \sqrt{\sum_{k=1}^{m} w_{jk}^2}}, \quad \text{i, j = 1, 2, ..., n.} \tag{4.2}$$

Em que $\sum_{k=1}^{m} w_{ik} w_{jk}$ corresponde ao produto interno dos vetores s_i e s_j, sendo w_{ik} e w_{jk} os pesos associados ao termo k nas sentenças s_i e s_j, respectivamente. Os termos $\sqrt{\sum_{k=1}^{m} w_{ik}^2}$ e $\sqrt{\sum_{k=1}^{m} w_{jk}^2}$ representam as normas ou magnitudes dos vetores s_i e s_j, respectivamente. A norma de um vetor é uma medida de seu "tamanho"ou "comprimento"no espaço vetorial. Ao multiplicar as normas dos vetores s_i e s_j, o produto interno é normalizado, garantindo que a medida de similaridade corresponda a uma razão da projeção vetorial, resultando em um valor entre -1 e 1.

5

Resumo Extrativo Baseado em Tópicos

Com as lacunas identificadas na literatura por meio do Mapeamento Sistemático e da revisão dos trabalhos relacionados, o presente capítulo se dedica à proposta de um novo método de Resumo Extrativo Baseado em Tópicos, abordando as limitações identificadas.

Este capítulo reproduz parcialmente o artigo *A Topic Based Generic Extractive Multi-document Text Summarization Method Using Memetic Algorithm and Combinatorial Optimization*, submetido ao periódico **Memetic Computing** no qual o método de resumo extrativo de texto intitulado *Maximum Relevance with Minimum Redundancy using the Shuffle Frog-Leaping Algorithm* (MRMRSFLA) é apresentado.

5.1 Principais Contribuições

Este trabalho concentra-se no problema de sumarização genérica extrativa de textos multi-documento. Neste estudo, o método *Maximum Relevance with Minimum Redundancy using Shuffle Frog-Leaping Algorithm* (MRMRSFLA) é desenvolvido, implementado e aplicado para resolver o problema de sumarização genérica extrativa de textos multi-documento. Os experimentos foram conduzidos utilizando os conjuntos de dados DUC 2001 e DUC 2002 (NIST, 2025), um *benchmark* público amplamente utilizado na literatura. Os documentos desses conjuntos de dados consistem em artigos jornalísticos em língua inglesa que abrangem diversos tópicos, provenientes de fontes como *Financial Times*, *Associated Press* e *The Wall Street Journal*. Para cada documento do conjunto de dados, há um resumo gerado por humanos (*gold standard*) e o documento original. Os resultados foram avaliados por meio da métrica *Recall-Oriented Understudy for Gisting Evaluation* (ROUGE) (LIN, 2004), uma medida padrão para a avaliação de resumos automáticos de texto (EL-KASSAS et al., 2021).

As limitações discutidas na Subseção 3.4.4 e na Seção 4 foram abordadas no MRMRSFLA, uma vez que o método foi avaliado de forma robusta por meio da definição de seus parâmetros utili-

zando o conjunto de dados DUC2001 e da mensuração de seu desempenho também no DUC2002 (Subseção 5.5), demonstrando sua capacidade de generalização. O método é uma extensão do *Shuffle Frog Leaping Algorithm* (Subseção 5.4.2), um algoritmo memético caracterizado por baixa complexidade computacional. O MRMRSFLA não depende de especialistas em nenhuma etapa do pipeline, é não supervisionado, foi projetado para ser genérico e é aplicável tanto a tarefas de sumarização de documento único quanto de múltiplos documentos, independentemente de domínio ou idioma. O MRMRSFLA também não descarta soluções candidatas que excedam um comprimento máximo predefinido; em vez disso, penaliza-as proporcionalmente ao excesso de comprimento, garantindo que soluções potencialmente adequadas não sejam descartadas devido a violações mínimas de tamanho. Ademais, o método emprega uma abordagem baseada em tópicos, cobrindo de forma eficaz a maioria dos subtópicos do documento.

Assim, as principais contribuições deste artigo podem ser resumidas da seguinte forma:

- O problema de sumarização genérica extrativa de textos multi-documento foi formulado como um problema de otimização combinatória quadrática inteira, envolvendo a otimização dos seguintes critérios: maximização da relevância das sentenças, minimização da redundância e maximização da informatividade das sentenças selecionadas.
- O MRMRSFLA, um algoritmo memético baseado em inteligência de enxames (*swarm*), foi desenvolvido para resolver esse problema pela primeira vez.
- Experimentos foram conduzidos utilizando os conjuntos de dados DUC 2001 e DUC 2002 e as métricas ROUGE.
- O MRMRSFLA supera os resultados de outras 6 abordagens no DUC2001 e de 10 abordagens no DUC2002 reportadas na literatura científica.

O restante deste artigo está organizado da seguinte maneira. A Seção 5.3 formula o problema de sumarização genérica de textos multi-documento como um problema de otimização combinatória quadrática inteira. A Seção 5.4 apresenta e descreve detalhadamente o MRMRSFLA. A Seção 5.5 apresenta a etapa de pré-processamento, os conjuntos de dados utilizados, as métricas de avaliação, a configuração de parâmetros, os resultados obtidos com a abordagem proposta e as comparações com outros métodos da literatura. Por fim, a Seção 5.6 conclui o artigo e discute direções para pesquisas futuras.

5.2 Modelagem de Tópicos

A modelagem de tópicos (agrupamento) foi implementada com o objetivo de determinar dinamicamente o número de tópicos, utilizando três métricas de avaliação: Davies-Bouldin, Calinski-Harabasz e Silhouette. Essas métricas foram implementadas por meio da biblioteca

scikit-learn (PEDREGOSA et al., 2011). Por padrão, foi definido um número máximo de 10 tópicos, e os modelos foram testados e avaliados para quantidades de tópicos variando de 2 a 10, com base nas três métricas. O número de tópicos foi selecionado por votação majoritária e, em caso de empate, optou-se pelo maior número de tópicos.

5.3 Formulação Matemática do Problema de Otimização

O problema de otimização pode ser formalmente definido da seguinte forma. Seja $D = d_1, d_2, d_3, \ldots, d_N$ uma coleção de documentos composta por N documentos. Alternativamente, a coleção pode ser expressa como $D = s_1, s_2, s_3, \ldots, s_n$, na qual consiste em um conjunto de n sentenças extraídas de todos os documentos. O objetivo deste problema é construir um resumo S por meio da seleção de um subconjunto de sentenças de D ($S \subset D$), respeitando os seguintes critérios:

- **Maximização da relevância**. O resumo deve incluir as sentenças mais relevantes de acordo com os tópicos centrais dos documentos.
- **Minimização da redundância**. O resumo deve evitar a inclusão de sentenças excessivamente semelhantes entre si.
- **Informatividade**. O resumo deve conter sentenças que forneçam o conteúdo mais informativo.
- **Restrição de comprimento**. O resumo deve possuir um comprimento predefinido L.

A maximização da relevância visa selecionar sentenças que sejam mais semelhantes ao tópico central, enquanto a redução da redundância busca evitar a inclusão de sentenças altamente similares no resumo candidato. A informatividade, por sua vez, promove a seleção de um conjunto de sentenças em que todas contribuam com informações relevantes. Para satisfazer esses critérios, a função objetivo proposta em Alguliyev, Aliguliyev e Isazade (2015) foi adaptada e aprimorada com o intuito de melhorar a seleção de sentenças informativas por meio da maximização da entropia sobre os valores de relevância (α), por meio da adição do termo de informatividade $\gamma H(X)$. Assim, a seguinte função objetivo foi formulada:

$$\text{maximize } f(X) = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \alpha_i \alpha_j (1 - \text{sim}(S_i, S_j)) x_i x_j + \gamma H(X), \tag{5.1}$$

$$\text{Subject to } \sum_{i=1}^{n} l_i x_i \leq L, \tag{5.2}$$

$$x_i \in \{0, 1\}, \text{ for } i = 1, \ldots, n \tag{5.3}$$

em que l_i representa o comprimento da sentença S_i. O número de palavras mede tanto o tamanho do resumo quanto o da sentença. Além disso, o termo *sim* representa a similaridade do cosseno entre as sentenças (S_i, S_j).

A relevância relativa das sentenças candidatas ao resumo foi computada da seguinte forma:

$$\alpha_i = \frac{\text{sim}(s_i, O)}{\sum_{j=1}^{n} \text{sim}(s_j, O)}, \text{i} = 1, 2, ..., \text{n} \tag{5.4}$$

em que O é o centro da coleção $D = \{s_1, s_2, ..., s_n\}$, e a (k-ésima coordenada o_k do centro é calculada como:

$$o_k = \frac{1}{n} \sum_{i=1}^{n} w_{i,k} \tag{5.5}$$

O peso definido na Equação (5.4) determina a relevância relativa da sentença s_i em relação ao conteúdo principal da coleção D.

A função objetivo (5.1) garante que o resumo capture de forma eficaz o conteúdo central do documento, mantendo, assim, sua relevância para o usuário. Isso é alcançado por meio do multiplicador $\alpha_i \cdot \alpha_j$, que reforça a seleção de sentenças altamente relevantes. Adicionalmente, a função evita a inclusão de sentenças que transmitam informações redundantes, minimizando a redundância. Tal propriedade é assegurada pelo multiplicador $1 - \text{sim}(S_i, S_j)$, que desencoraja a seleção de sentenças altamente similares. Para promover a informatividade, a entropia $(\gamma H(X))$ é computada utilizando o vetor de relevância (α) associado às sentenças selecionadas, de modo a medir sua distribuição de probabilidade. O parâmetro γ controla a importância relativa da informatividade, possui valor padrão igual a 0,5 e não foi incluído entre os parâmetros a serem otimizados.

Na Equação 5.1, Com o objetivo de aumentar a informatividade (*informativeness*), a entropia $(\gamma H(X))$ é calculada com base no vetor de relevância (α) correspondente às sentenças selecionadas, o qual representa sua distribuição de probabilidade.

A entropia atua como uma medida da diversidade informacional no conjunto de sentenças selecionadas. Um valor de entropia mais elevado indica uma distribuição mais equilibrada e heterogênea dos escores de relevância, sugerindo que o resumo é composto por conteúdos variados e informativos, sem que uma única sentença influencie de forma desproporcional o resumo. Essa maior incerteza reflete uma cobertura mais ampla das informações relevantes (KHURANA; BHATNAGAR, 2022).

Ao integrar a entropia ao processo de seleção, o método promove uma escolha equilibrada de sentenças com diferentes níveis de relevância, melhorando, assim, a informatividade geral do resumo. A entropia é calculada pela Eq. 6.7, de acordo com Shannon (1948).

$$h = -\sum p_i \cdot \log_2(p_i) \tag{5.6}$$

A restrição (5.2) limita o comprimento do resumo, enquanto a Equação (5.3) impõe a restrição de integralidade sobre x_i, a qual é automaticamente satisfeita na formulação do problema considerada.

Para avaliar a qualidade de um resumo candidato, é necessária uma função de aptidão (*fitness*), uma vez que seu valor atua como um indicador de quão adequadamente o resumo atende ao problema de otimização. Conforme Alguliyev, Aliguliyev e Isazade (2015), um termo de penalidade foi incorporado à função de aptidão com o objetivo de transformar o problema restrito em um problema irrestrito.

Um termo de penalidade adicional é introduzido para desencorajar soluções inviáveis, por meio da aplicação de um fator de penalidade $\beta(\beta > 0)$. A função (*fitness*) é formalmente definida da seguinte forma:

$$fit(X) = f(X) \cdot exp\left(-\beta \cdot max\left(0, \sum_{i=1}^{n} l_i \cdot x_i - L\right)\right), \tag{5.7}$$

O primeiro termo, $f(X)$ em (5.7), corresponde à função objetivo definida em (5.1). O segundo termo introduz uma função de penalidade para maximização, na qual β representa o custo de penalidade associado ao excedente do comprimento pré-definido do resumo. O valor inicial de β é definido pelo usuário. Caso a solução seja inviável, o termo de penalidade reduz o valor da aptidão, direcionando a busca para soluções factíveis. Por outro lado, se a restrição de comprimento do resumo for satisfeita, esse termo assume valor igual a um, garantindo que a solução não seja penalizada.

Para impor dinamicamente a factibilidade, β pode ser progressivamente incrementado ao longo da execução, aplicando, assim, um controle adaptativo do custo de penalidade. Essa adaptação é expressa por $\beta = \beta^- + (\beta^+ - \beta^-)t/t_{max}$, em que t_{max} representa o número máximo de iterações, e β^- e β^+ denotam, respectivamente, os valores inicial e final do parâmetro de penalidade. Neste estudo, adotou-se $\beta^- = 0.1$ e $\beta^+ = 0.9$.

5.4 Maximum Relevance with Minimum Redundancy using Shuffle Frog-Leaping Algorithm

Nesta seção, o *Maximum Relevance with Minimum Redundancy using Shuffle Frog-Leaping Algorithm* (MRMRSFLA) é apresentado. Primeiramente, o algoritmo SFLA básico é descrito. Em seguida, as etapas de preprocessamento são definidas. E, por fim, as principais etapas do MRMRSFLA e seus principais operadores são explicados.

5.4.1 Algoritmo Base

A meta-heurística memética *Shuffled Frog-Leaping* foi proposta por Eusuff, Lansey e Pasha (2006) para resolver problemas de otimização combinatória. A SFLA é uma metáfora de busca cooperativa baseada na população inspirada na memética natural. Ele consiste em particionar a população de soluções candidatas (sapos) em subconjuntos, chamados de *memeplexes*, em que os indivíduos interagem entre si. Os sapos virtuais agem como hospedeiros ou portadores de memes, onde um meme é uma unidade de evolução cultural. O algoritmo performa simultaneamente uma busca local independente em cada *memeplexe*, e uma exploração global, já que os sapos são periodicamente embaralhados e reorganizados em novos *memeplexes*. O SFLA é apresentado em Algoritmo 1:

Algorithm 1 Pseudocódigo do SFLA

```
Population ← init_population(popsize)
Population ← calculate_fitness(Population)
Population ← sort_by_fitness(Population)
while ¬stop_criteria do
    Memplexes ← divide_pop_into_memplexes(Population, memnum)
    Memplexes ← local_search(Memplexes, memnum, improvsmax)
    Population ← combine_evolved_memplexes(Memplexes)
    Population ← calculate_fitness(Population)
    Population ← sort_by_fitness(Population)
end while=0
```

Do ponto de vista computacional, o SFLA apresenta uma complexidade que cresce de forma linear com o número de iterações e quase linear com o tamanho da população. Isso se deve principalmente às operações de ordenação e avaliação de *fitness*, que são executadas em cada ciclo iterativo. A etapa de busca local adiciona um fator de custo proporcional ao número de melhorias realizadas em cada *memeplex* (L), o que pode aumentar significativamente o tempo total de execução, especialmente em problemas em que a exploração local é intensiva. De modo geral, a complexidade total $O(I \cdot P(\log P + L))$ evidencia o equilíbrio entre exploração global (dependente de $P \log P$) e exploração local (dependente de LP). Em que P representa o tamanho da população ($popsize$), M o Número de memplexes ($memnum$), I o número máximo de iterações (ou até o critério de parada) e L representa o número máximo de melhorias locais por *memeplex* ($improvsmax$). Assim, a eficiência do SFLA está diretamente associada à calibração adequada dos parâmetros de população, iteração e intensidade de busca local, de modo a balancear o custo computacional e a qualidade da solução obtida. A Tabela 15 descreve a complexidade das operações do SFLA.

5.4.2 Principais etapas do MRMRSFLA

O algoritmo implementado neste trabalho é uma extensão do SFLA básico, adaptando-o para resumos genéricos e incorporando uma etapa de modelagem de tópicos. As etapas

Tabela 15 – Complexidade de tempo assintótica de cada procedimento SFLA.

Procedure	**Time Complexity**
`init_population(popsize)`	$O(P)$
`calculate_fitness(Population)`	$O(P)$
`sort_by_fitness(Population)`	$O(P \log P)$
`divide_pop_into_memplexes(Population, memnum)`	$O(P)$
`local_search(Memplexes, memnum, improvsmax)`	$O(LP)$
`combine_evolved_memplexes(Memplexes)`	$O(P)$
Per iteration (loop body)	$O(P \log P + LP)$
Overall algorithm	$O(I \cdot P(\log P + L))$

do MRMRSFLA são detalhadas no Algoritmo 2. A implementação do MRMRSFLA foi disponibilizada em um repositório público na plataforma GitHub[1].

Ele foi selecionado devido à baixa complexidade envolvida em sua adaptação para uma abordagem de sumarização genérica. Além disso, apresenta a vantagem de realizar buscas locais e globais para otimizar a função objetivo (5.1) de maneira relativamente simples.

A modificação necessária para essa transformação é detalhada na Seção 6.4.3. Além disso, uma nova etapa de pré-processamento envolvendo modelagem de tópicos foi incorporada ao algoritmo, conforme descrito na Seção 5.2.

Primeiro, os tópicos do *corpus* são definidos conforme descrito na Seção 5.2. Então, no primeiro *loop*, todas as operações são executadas para cada tópico dentro do conjunto de tópicos. Em seguida, a população inicial de resumos de candidatos é inicializada de forma aleatória e uniforme. Um resumo candidato é representado como um vetor binário e é gerado da seguinte forma:

$$x_i \sim \text{Bernoulli}(p = 0.5), \text{ para } i = 1, 2, \ldots, n \tag{5.8}$$

Em que x_i é o i-ésimo elemento do vetor binário candidato. A distribuição Bernoulli indica que cada x_i tem duas possíveis saídas (0 ou 1), com probabilidade $p = 0.5$, o que significa que cada sentença tem uma probabilidade igual de ser 0 ou 1, para todas as sentenças do total de sentenças candidatas n no vetor binário.

Posteriormente, os valores da função objetivo (5.7) são calculados para cada indivíduo, e a população é ordenada em ordem decrescente com base na aptidão. As operações do segundo loop são repetidas até que o número máximo de ciclos seja atingido, onde *cyclesmax* serve como critério de parada para o algoritmo. As operações executadas em cada ciclo impulsionam a evolução da população. Em cada ciclo, o melhor indivíduo global (X_{bestG}) é selecionado, e a população é dividida em *memeplexes memesnum*. Os indivíduos são distribuídos em *memeplexes* usando uma alocação pareada das soluções classificadas, o que significa que os indivíduos são atribuídos sequencialmente a diferentes *memeplexes* (o primeiro indivíduo é atribuído ao

[1] https://github.com/k3ybladewielder/mrmrsfla

Algorithm 2 Pseudocódigo do MRMRSFLA

```
Topics ← define_main_topics(corpus)
for topic to topics do
    Population ← init_population(popsize)
    Population ← calculate_objective_functions(Population, popsize)
    Population ← sort_by_fitness(Population, popsize)
    for cycle = 1 to cyclesmax do
        X_bestG ← select_best_global(Population)
        Memplexes ← divide_pop_into_memplexes(Population, memnum)
        for m = 1 to memesnum do
            for i = 1 to improvsmax do
                X_bestL ← select_best_local(Memplexes[m])
                X_worstL ← select_worst_local(Memplexes[m])
                save_worst_local(Population, X_worstL)
                X_new ← mutate_solution(X_bestL, p_m)
                if X_new ≻ X_worstL then
                    save_solution(Memplexes[m], X_new)
                else
                    X_new ← mutate_solution(X_bestG, p_m)
                    if X_new ≻ X_worstL then
                        save_solution(Memplexes[m], X_new)
                    else
                        X_new ← random_solution()
                        save_solution(Memplexes[m], X_new)
                    end if
                end if
                Memplexes[m] ← sort_by_dominance(Memplexes[m])
            end for
        end for
        Population ← combine_evolved_memplexes(Memplexes)
        Population ← calculate_objective_functions(Population, popsize * 2)
        Population ← sort_by_fitness(Population, popsize * 2)
        Savebest_individual, best_fitness, best_sentences
    end for
end for=0
```

primeiro memeplex, o segundo ao segundo e assim por diante) até que todos os indivíduos sejam distribuídos. Se o número máximo de *memeplexes* for atingido, a alocação recomeça a partir do primeiro *memeplexe* até que todos os indivíduos sejam atribuídos.

Em seguida, as operações incluídas no terceiro e quarto loop são executadas para cada *memeplexe* durante o número máximo de melhorias *improvemax* por *memeplexe* (quarto loop). As operações do quarto loop performam uma busca local em cada *memeplexe*. A cada etapa de melhoria, o melhor (X_{bestL}) e o pior (X_{worstL}) indivíduo local (do *memeplexe*) são armazenados. Em seguida, o processo de mutação é realizado no melhor local (X_{bestL}) e se a nova solução (X_{new}) for melhor que o pior local (X_{worstL}), ela é adicionada ao *memeplexe*. Se a nova solução não for melhor que o pior local, o processo de mutação é realizado com o melhor global (X_{bestG}) e se o novo indivíduo for melhor que o pior local, ele é adicionado ao *memeplexe*. Caso não seja melhor, então um indivíduo aleatório é gerado e adicionado ao *memeplexe*. O processo de mutação é explicado detalhadamente na Subseção 6.4.3.

Após finalizar a busca local para todos os *memeplexes* até o critério *improvemax*, os resultados envolvendo todos os *memeplexes* são recombinados e a função objetivo de *fitness* é calculada novamente para cada indivíduo, e a população é ordenada em ordem decrescente de

acordo com o fitness. Ao final de cada ciclo, um número *popsize* de indivíduos é selecionado como sobreviventes, e estes serão utilizados no próximo ciclo.

Em relação a complexidade, ela é fortemente influenciada pela estrutura hierárquica de repetição (por tópicos e ciclos) e pela intensidade da busca local realizada em cada *memeplex*. A complexidade assintótica total é dada por $O(T \cdot C \cdot P(\log P + M \cdot L))$. Em que T é o número de tópicos; C é o número máximo de ciclos (iterações globais); P é o tamanho da população; M é o número de memplexes (subgrupos); e L é o número máximo de melhorias locais ($improvsmax$).

A presença de laços aninhados sobre tópicos, ciclos e memplexes torna o custo computacional mais elevado em relação ao SFLA tradicional (Tabela 15). As operações mais custosas permanecem sendo as de avaliação das funções objetivo e ordenação por fitness, ambas com custo proporcional a $O(P \log P)$ por ciclo. A fase de busca local, repetida $M \cdot L$ vezes, adiciona um custo adicional linear em relação ao número de indivíduos e às tentativas de aprimoramento local.

Em termos qualitativos, a complexidade do MRMRSFLA depende diretamente de parâmetros como número de tópicos e o nível de exploração local (L). Essa complexidade é, no entanto, justificada pela maior capacidade de exploração multiobjetiva e multi-tópica do espaço de busca, o que tende a gerar soluções mais robustas e diversificadas. A Tabela 16 descreva a complexidade das operações do MRMRSFLA.

Tabela 16 – Complexidade de tempo assintótica de cada procedimento MRMRSFLA.

Procedimento	**Complexidade**
`define_main_topics(corpus)`	$O(T)$
`init_population(popsize)`	$O(P)$
`calculate_objective_functions(Population)`	$O(P)$
`sort_by_fitness(Population)`	$O(P \log P)$
`divide_pop_into_memplexes(Population, memnum)`	$O(P)$
`local_search (nested over` M`,` L`)`	$O(M \cdot L \cdot P/M) = O(LP)$
`mutate_solution()`	$O(1)$
`combine_evolved_memplexes(Memplexes)`	$O(P)$
`calculate_objective_functions(Population, popsize * 2)`	$O(P)$
`sort_by_dominance / fitness()`	$O(P \log P)$
Per cycle (for each topic)	$O(P(\log P + L))$
Per topic (across all cycles)	$O(C \cdot P(\log P + L))$
Overall algorithm (all topics)	$O(T \cdot C \cdot P(\log P + L))$

5.4.3 Mutação

O operador de mutação proposto em Sanchez-Gomez, Vega-Rodríguez e Pérez (2022) foi adotado com modificações. Como o método apresentado neste estudo consiste em uma abordagem de sumarização genérica, todas as comparações são realizadas com o centróide, e não com a consulta, conforme originalmente proposto.

O processo de mutação envolve a adição, remoção ou substituição de uma sentença no resumo candidato (indivíduo), de acordo com o tipo de mutação selecionado. Cada uma das três alternativas de mutação possui a mesma probabilidade de ser escolhida, sendo aplicada apenas uma por operação de mutação. A probabilidade de mutação é definida como $p_m = 1/n$, em que n representa o número total de sentenças, garantindo que apenas uma única sentença seja submetida à mutação em cada iteração. A mutação ocorre sempre, independentemente de a sentença modificada contribuir ou não para a melhoria da solução.

A operação de adição insere uma sentença, proveniente do conjunto de sentenças disponíveis, que ainda não esteja presente no resumo candidato. A sentença recém-adicionada ($s_i \notin S$) deve aprimorar a qualidade do resumo candidato. Em outras palavras, a similaridade do cosseno da nova sentença com o centróide deve ser superior à similaridade média de todas as sentenças do indivíduo em relação ao centróide:

$$cosim(s_i, O) > \frac{1}{n} \sum_{j=1}^{n} cosim(s_j, O). \tag{5.9}$$

A sentença $s_i \notin S$ é selecionada aleatoriamente do conjunto de documentos D, e se cumprir as condições, será adicionada ao resumo candidato. Caso não cumpra, a próxima sentença $s_i \notin S$ com maior similaridade de cosseno com o centro será adicionada.

A remoção de sentença faz com que uma das sentenças do resumo candidato seja descartada. A sentença que será descartada não deve deteriorar a qualidade do resumo candidato. Sendo assim, a similaridade de cosseno da sentença $s_i \in S$ com o centro deve ser menor que a média da similaridade de cosseno de todas as sentenças com o centro.

$$cosim(s_i, O) < \frac{1}{n} \sum_{j=1}^{n} cosim(s_j, O). \tag{5.10}$$

Assim como na adição, a sentença $s_i \in S$ é selecionada aleatoriamente do resumo candidato, e, se cumpridas as condições, ela é removida do resumo. Caso não seja cumprida, a próxima sentença $s_i \in S$ é checada até que a operação seja executada. Se nenhuma sentença cumprir as condições, então a sentença $s_i \notin S$ com menor similaridade de cosseno será removida.

A operação de substituição de frase troca uma frase da coleção de documentos D que não está no resumo por uma que já está incluída. Neste caso, o processo de mutação retira uma frase do resumo e a substitui por outra diferente da coleção.

5.5 Resultados Experimentais

Nesta seção, são descritos os *datasets* utilizados, a definição de parâmetros, os resultados obtidos com a abordagem proposta e a comparação com os resultados da literatura.

5.5.1 Preprocessamento

Antes de executar o algoritmo, algumas etapas de pré-processamento devem ser realizadas nos documentos da coleção D. Essas etapas são as seguintes:

- **Segmentação**: cada frase do documento deve ser extraída individualmente, definindo claramente seu início e fim.
- **Tokenização**: todas as palavras em uma frase são tokenizadas para remover caracteres especiais, pontos de interrogação e pontuação.
- **Remoção de palavras irrelevantes**: palavras que não carregam significado semântico significativo e aparecem com frequência, como preposições, conjunções e artigos, são excluídas das frases. A operação de remoção de palavras irrelevantes foi realizada usando a biblioteca NLTK[2].
- **Lematização**: a raiz de cada palavra foi extraída usando o módulo SnowballStemmer da biblioteca NLTK.
- **Representação**: as palavras nas frases são representadas numericamente através da aplicação do TF-ISF.

5.5.2 Bases de Dados

Os resultados do algoritmo proposto foram avaliados nos conjuntos de dados DUC2001 e DUC2002, ambos conjuntos de dados de referência disponíveis publicamente, fornecidos pela *Document Understanding Conference* (DUC) (NIST, 2025) para avaliação automática de resumo de texto. Os documentos nesses conjuntos de dados consistem em artigos de notícias em inglês que cobrem vários tópicos de fontes como *Financial Times*, *Associated Press* e *The Wall Street Journal*. DUC2001 e DUC2002 consistem em 30 e 59 tópicos, respectivamente, e contêm resumos de 100 palavras gerados por humanos para cada documento. O DUC2001 é composto por 302 documentos, enquanto o DUC2002 é composto por 533 documentos que abrangem diversos assuntos. Para cada documento do conjunto de dados, há um resumo gerado por humanos (padrão ouro) e o documento original. A tabela 17 fornece uma descrição dos conjuntos de dados DUC2001 e DUC2002.

Na média, os resumos de referência para o *dataset* DUC2002 são maiores, tanto em número de sentenças por tópico e por documentos, assim como o número de palavras por tópico e por documento. Todos os documentos foram segmentados em sentenças e pré-processados. O processo de pré-processamento dos dados é descrito na Seção 5.5.1.

[2]https://www.nltk.org/

Tabela 17 – Descrição dos conjuntos de dados DUC2001 e DUC2002.

	DUC2001	DUC2002
Número de Tópicos	30	59
Número de Documentos	302	533
Número médio de Sentenças por Tópico	59.5	150.55
Número médio de Sentenças por Documento	5.91	15.55
Número médio de Termos por Tópico	1012.97	2138.68
Número médio de Termos por Documento	100.62	236.74

5.5.3 Definição de Parâmetros

Para definir os parâmetros, foi realizado um teste utilizando uma amostra do DUC2001. A amostra foi determinada com nível de confiança de 0,95 no cálculo do tamanho amostral, resultando em 169 (55%) exemplos. O teste envolveu a execução do algoritmo por 5 e 10 ciclos (*cyclesmax*), selecionando aleatoriamente valores para o número de iterações (*improvsmax*) e o número de *memeplexes* (*memesnum*), e medindo seu desempenho usando as métricas descritas na subseção 7.4.3. Os valores possíveis para o número de iterações (*improvsmax*) foram 25, 50, 75 e 100, enquanto os valores possíveis para o número de *memeplexes* (*memesnum*) foram definidos como 5, 10, 15, 20 e 25. O teste foi realizado independentemente 31 vezes para cada parâmetro do ciclo (*cyclesmax*), e em cada execução do teste os parâmetros foram selecionados aleatoriamente, com cada um tendo igual probabilidade de ser escolhido. A tabela 18 descreve os espaços de busca para os parâmetros otimizados. Os resultados médios detalhados para todos os parâmetros testados são apresentados na Tabela 19.

Tabela 18 – Espaços de busca para os parâmetros de otimização do algoritmo

Parâmetro	**Valores de busca**
Número de ciclos (*cyclesmax*)	$\{5, 10\}$
Número de iterações (*improvsmax*)	$\{25, 50, 75, 100\}$
Número de *memeplexes* (*memesnum*)	$\{5, 10, 15, 20, 25\}$
Nível de confiança (cálculo do tamanho da amostra)	0.95
Tamanho da amostra (exemplos do DUC2001)	169 (55% do conjunto de dados)
Repetições independentes de teste	31 execuções por valor de *cyclesmax*

Tabela 19 – Resultados médios experimentais no conjunto de dados DUC2001 com diferentes configurações de parâmetros.

No. Cycle	No. Iterations	No. Memeplex	ROUGE-1 Precision	ROUGE-2 Precision	ROUGE-1 Recall	ROUGE-2 Recall	ROUGE-1 F1	ROUGE-2 F1
5	25	5	0.202874	0.082965	0.449924	0.179325	0.263981	0.107141
5	25	10	0.204238	0.085172	0.448849	0.179315	0.265521	0.109510
5	25	15	0.201960	0.081477	0.440167	0.171897	0.261589	0.104616
5	25	25	0.199330	0.079816	0.435976	0.169529	0.257905	0.102567
5	50	5	0.202514	0.082026	0.447789	0.176290	0.263095	0.105882
5	50	10	0.203372	0.083389	0.443199	0.175244	0.263383	0.107085
5	50	15	0.205957	0.083990	0.445166	0.177569	0.265724	0.107980
5	50	20	0.203245	0.083446	0.444402	0.175476	0.263067	0.106895
5	50	25	0.202874	0.082625	0.439951	0.174264	0.262448	0.106312
5	75	5	0.205190	0.084064	0.449676	0.178744	0.265845	0.108249
5	75	10	0.202330	0.082130	0.444708	0.176106	0.262664	0.105976
5	75	20	0.205269	0.083346	0.447389	0.176890	0.266260	0.107309
5	75	25	0.201451	0.082749	0.437541	0.175722	0.260933	0.106801
5	100	5	0.202675	0.082422	0.447198	0.177076	0.263534	0.106635
5	100	10	0.202170	0.082128	0.441226	0.173491	0.262126	0.105693
5	100	15	0.203889	0.083292	0.438778	0.173686	0.263199	0.106787
10	25	5	0.201926	0.082577	0.449667	0.179171	0.263212	0.107105
10	25	10	0.203794	0.082692	0.445054	0.175115	0.263427	0.106141
10	25	20	0.204910	0.084932	0.449093	**0.181264**	0.265570	0.109556
10	25	25	0.202207	0.082255	0.439436	0.176628	0.261342	0.106303
10	50	5	0.200486	0.082693	0.436843	0.172529	0.259521	0.105915
10	50	10	0.202657	0.081384	0.443005	0.171783	0.262597	0.104616
10	50	20	0.205995	0.083313	**0.450919**	0.178111	0.266828	0.107450
10	50	25	0.204591	0.084259	0.442762	0.179279	0.263812	0.108417
10	75	5	0.201179	0.080949	0.444354	0.171359	0.260521	0.103512
10	75	10	**0.207954**	0.085788	0.447769	0.179168	**0.268355**	0.109737
10	75	20	0.205871	0.084065	0.443428	0.175338	0.265440	0.107500
10	100	5	0.203658	0.081679	0.443808	0.174072	0.263207	0.105066
10	100	10	0.202976	0.080882	0.439287	0.171766	0.262110	0.104107
10	100	15	0.199812	0.077660	0.430722	0.164180	0.257897	0.099786
10	100	20	0.207338	**0.085893**	0.444283	0.178239	0.267088	**0.109794**

Após a realização de 31 execuções para cada documento da amostra calculada, foram geradas aleatoriamente um total de 31 configurações de parâmetros.

Para verificar se havia diferença significativa entre o uso de 5 e 10 ciclos, foi aplicado um teste de hipótese não paramétrico de Wilcoxon Signed-Rank (pareado). Foram formados dois grupos: o Grupo 1 com *cyclesmax* = 5 e o Grupo 2 com *cyclesmax* = 10, ambos contendo 5239 observações. O teste resultou em um p-valor igual a 0.820646 e uma estatística de Wilcoxon de 5243241.0000, indicando a ausência de evidências para rejeitar a hipótese nula de que as distribuições de *Recall* do ROUGE-1 são equivalentes. A variação do parâmetro *cyclesmax* não impactou significativamente o ROUGE-1 (Recall), sugerindo que o aumento do número de ciclos não produziu efeito claro sobre a métrica avaliada.

Após essa análise, todas as combinações de parâmetros *(cyclesmax–improvsmax–memesnum)* foram testadas utilizando o **teste de Friedman**, um teste estatístico não paramétrico empregado para detectar diferenças entre múltiplas amostras relacionadas. Esse teste é uma extensão do teste de Wilcoxon Signed-Rank e é particularmente adequado para a comparação de vários algoritmos ou configurações de modelos avaliados sobre os mesmos conjuntos de dados ou unidades experimentais. Neste experimento, diferentes configurações de parâmetros foram avaliadas sobre o mesmo conjunto de documentos.

O teste de Friedman ordena o desempenho de cada versão (configuração) dentro de cada documento, atribuindo a classificação 1 à configuração de melhor desempenho. Em seguida, avalia se a distribuição dessas classificações difere significativamente entre as configurações. Diferentemente da ANOVA, o teste não assume normalidade dos dados, o que o torna apropriado para métricas de avaliação como o ROUGE.

Os resultados obtidos foram os seguintes: a estatística qui-quadrado ($\chi^2 = 36.94$), com ($k - 1 = 30$) graus de liberdade, produziu um p-valor igual a 0.1788, superior ao nível de significância adotado ($\alpha = 0.05$). Dessa forma, a hipótese nula não pôde ser rejeitada, indicando que não há diferença estatisticamente significativa entre as 31 configurações de parâmetros avaliadas. O coeficiente de Kendall's W, que mede o grau de concordância entre os *rankings* (variando de 0, para ausência de concordância, a 1, para concordância perfeita), foi igual a 0.0073, um valor extremamente baixo. Isso indica que os *rankings* de desempenho das configurações foram amplamente inconsistentes entre os documentos, não havendo uma configuração que superasse consistentemente as demais.

Em termos práticos, esses resultados indicam que as variações no conjunto de parâmetros *(cyclesmax–improvsmax–memesnum)* não resultaram em melhorias significativas ou sistemáticas no *Recall* do ROUGE-1. Embora pequenas flutuações tenham sido observadas entre as configurações, tais diferenças são provavelmente decorrentes de variação aleatória, e não de um efeito real das alterações nos parâmetros. Diante do resultado não significativo do teste global de Friedman, não foram realizados testes pós-hoc pareados (por exemplo, o teste de Nemenyi), uma vez que não houve evidência de que alguma configuração fosse estatisticamente superior às demais.

Por fim, os parâmetros adotados corresponderam àqueles que apresentaram o maior valor médio para a métrica de avaliação ROUGE-1 *Recall*, a saber, *cyclesmax* = 10, *memesnum* = 20 e *improvsmax* = 50, conforme apresentado na Tabela 19.

5.5.4 Resultados com o Método Proposto

Nesta seção, os resultados obtidos utilizando o MRMRSFLA são apresentados e analisados. Foi realizada uma análise estatística com os resultados obtidos para ROUGE-1 e ROUGE-2 para todos os tópicos dos *datasets* utilizados.

As tabelas incluem o valor médio, mediano e desvio padrão, o primeiro e terceiro quartil (Q1 e Q3), e o valor mínimo e valor máximo para os *scores* ROUGE baseado em 31 repetições por amostra em execuções independentes para ambos os *datasets* DUC2001 e DUC2002.

Os resultados apresentados nas Tabelas 20 e 21 mostram os resultados médios das métricas ROUGE-1 e ROUGE-2 obtidas pelo MRMRSFLA. Estes valores médios serão utilizados nas comparações na subseção a seguir.

Tabela 20 – Resultados obtidos pelo MRMRSFLA para ROUGE-1 e ROUGE-2 utilizando o DUC2001

MRMRSFLA	ROUGE-1 Precision	ROUGE-2 Precision	ROUGE-1 Recall	ROUGE-2 Recall	ROUGE-1 F1	ROUGE-2 F1
Média	0.193391	0.086548	**0.544442**	0.243031	0.269673	0.121007
Mediana	0.17898	0.070234	0.547945	0.212444	0.26087	0.101523
Desvio Padrão	0.087675	0.068053	0.190153	0.165698	0.100625	0.088786
Q1	0.127025	0.037838	0.396552	0.116667	0.196488	0.057471
Q3	0.247778	0.116959	0.684932	0.327869	0.329376	0.164948
Mínimo	0	0	0	0	0	0
Máximo	0.571429	0.495327	1	0.956522	0.681564	0.609137

Tabela 21 – Resultados obtidos pelo MRMRSFLA para ROUGE-1 e ROUGE-2 utilizando o DUC2002

MRMRSFLA	ROUGE-1 Precision	ROUGE-2 Precision	ROUGE-1 Recall	ROUGE-2 Recall	ROUGE-1 F1	ROUGE-2 F1
Média	0.226975	0.105661	**0.560506**	0.262218	0.309174	0.144199
Mediana	0.215962	0.089552	0.559322	0.234375	0.298643	0.125786
Desvio Padrão	0.085269	0.06946	0.189483	0.160224	0.098131	0.088748
Q1	0.1625	0.055794	0.424658	0.140625	0.23913	0.080402
Q3	0.282353	0.140351	0.69697	0.354839	0.369942	0.190476
Mínimo	0.012195	0	0.015385	0	0.014286	0
Máximo	0.679245	0.596154	1	0.936508	0.661765	0.614583

Para o DUC2001, a abordagem proposta o primeiro e terceiro quartil se concentram entre o valor de 0.4 e 0.7 respectivamente. Já os resultados para DUC2002, o desempenho dos valores centrais foi relativamente mais baixo se comparado com o DUC2001, mas destacam-se os desempenhos *outliers*, em que um subgrupo de documentos obtiveram um resultado muito alto em relação aos demais. Na média, o valor para DUC2002 foi levemente superior que o DUC2001. As Figura 9 e 10 incluem o *boxplot* os histogramas dos resultados obtidos para ROUGE-1 e ROUGE-2.

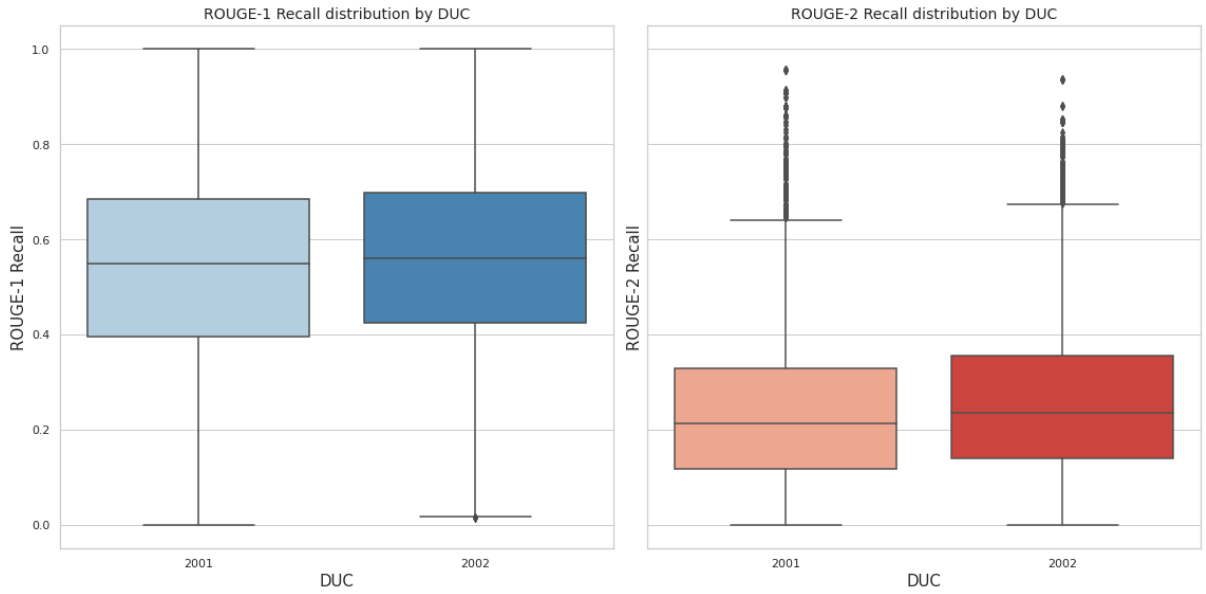


Figura 9 – *Boxplot*s obtidos pelo MRMRSFLA para ROUGE-1, ROUGE-2 (*Recall*).

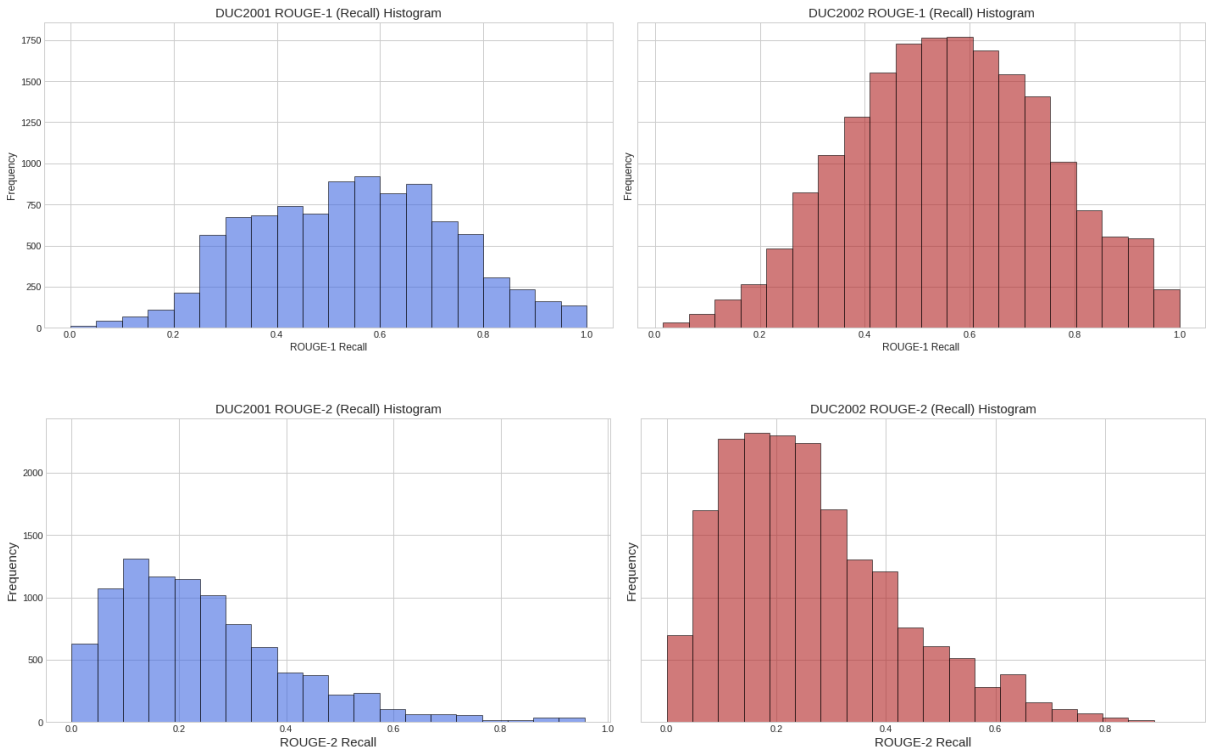


Figura 10 – Histogramas obtidos pelo MRMRSFLA para ROUGE-1 e ROUGE-2 (*Recall*).

5.5.5 Comparação com resultados de outras abordagens

A subseção a seguir apresenta os resultados obtidos por outras abordagens, as quais serão comparadas com os resultados da metodologia proposta neste trabalho.

As tabelas 22 e 23 mostram os resultados comparativos para os conjuntos de dados DUC2001 e DUC2002 com base nas métricas avaliadas para cada método concorrente. As métricas relatadas dos métodos em comparação com MRMRSFLA foram relatadas pelos autores

em seus trabalhos, além disso, eles usaram os mesmos conjuntos de dados para definição de parâmetros. Eles relataram o *Recall* médio para ROUGE-1 e ROUGE-2, bem como (entre parênteses) a melhoria percentual alcançada por MRMRSFLA, calculada (mrmrsfla_result - alternativa_resultado)/alternativa_resultado) * 100). O hífen (-) foi utilizado quando o resultado de uma determinada métrica não está disponível.

Tabela 22 – Comparação do modelo proposto com outros métodos usando ROUGE-1 e ROUGE-2 no DUC2001 **(Recall)**

Métodos	**ROUGE -1**		**ROUGE-2**	
MRMRSFLA	**0.544442**	-	**0.243031**	-
FEC without Gap	0.4632	(+9.52)	0.2112	(+15.07)
FEC_B&B	0.4971	(+11.90)	0.2081	(+16.78)
FEC_Gap	0.4865	(+13.05)	0.2054	(+18.32)
FEC_GAP*	0.4816	(+17.54)	0.1873	(+29.75)
FEC_WGAP	0.4621	(+17.82)	0.1866	(+30.24)
FEC_WGAP*	0.4613	(+18.02)	0.1831	(+32.73)
Average others	0.4753	(+14.55)	0.19695	(+23.38)

Tabela 23 – Comparação do modelo proposto com outros métodos usando ROUGE-1 e ROUGE-2 no DUC2002 **(Recall)**

Métodos	**ROUGE -1**		**ROUGE-2**	
MRMRSFLA	**0.560506**		**0.262218**	
FbTS	0.43803	(+27.96)	0.21212	(+23.62)
FUZZY CST with COM	0.33206	(+68.80)	0.12806	(+104.76)
FEC without Gap	0.4637	(+20.88)	0.1889	(+38.81)
FEC_B&B	0.4987	(+12.39)	0.2187	(+19.89)
FEC_Gap	0.4613	(+21.50)	0.2163	(+21.23)
FEC_GAP*	0.4634	(+20.95)	0.1966	(+33.38)
FEC_WGAP	0.4636	(+20.90)	0.2138	(+22.65)
FEC_WGAP*	0.4613	(+21.50)	0.1882	(+39.33)
Punctuation+Lemma Stemming	0.352	(+59.23)	-	-
Punctuation+Root Stemming	0.4572	(+22.59)	-	-
Average others	0.439129	(+27,64)	0.195335	(+34.24)

Todos os métodos incluídos na comparação foram descritos na Capítulo 4. Para o DUC2001, os métodos de última geração utilizados para comparação incluem *Branch and Bound* (B&B), estatísticas de lacunas padrão (Gap), estatísticas de lacunas sem função logarítmica (Gap*), estatísticas de lacunas ponderadas (WGap) e estatísticas de lacunas ponderadas sem função logarítmica (WGap*), conforme proposto por Verma, Verma e Pal (2022). Para DUC2002, as comparações foram feitas com os métodos FbTS (TOMER; KUMAR, 2022), FUZZY CST com COM (KUMAR et al., 2014), FEC sem Gap (VERMA; VERMA; PAL, 2022), FEC_B&B (VERMA; VERMA; PAL, 2022), FEC_Gap (VERMA; VERMA; PAL, 2022), FEC_GAP* (VERMA; VERMA; PAL, 2022), FEC_WGAP (VERMA; VERMA; PAL, 2022), FEC_WGAP* (VERMA; VERMA; PAL, 2022), pontuação+*lema stemming* (ALQAISI; GHANEM; QAROUSH, 2020), e pontuação+*root stemming* (ALQAISI; GHANEM; QAROUSH, 2020).

Nos *benchmarks* DUC2001 e DUC2002, o método proposto, MRMRSFLA, superou os métodos de última geração. No DUC2001, ele apresentou um aumento de 9.52% e 15.07% no *Recall* médio do ROUGE-1 e ROUGE-2, respectivamente, em comparação com o melhor método existente. Quando comparado ao pior método, a melhoria foi ainda mais expressiva, atingindo 18.02% no ROUGE-1 e 32.73% no ROUGE-2.

Já no DUC2002, o MRMRSFLA obteve um ganho de 12.39% e 21.23% no ROUGE-1 e ROUGE-2, respectivamente, em relação ao melhor método. Em comparação com os piores métodos, a melhoria foi ainda mais significativa, alcançando 68.80% no ROUGE-1 e impressionantes 104.76% no ROUGE-2.

Na média, a abordagem proposta resultou numa melhoria percentual de 14,5% no ROUGE-1 e 23,38% no ROUGE-2 para o conjunto de dados DUC2001. Enquanto que para o DUC2002, obteve uma melhoria percentual média de 27,64% no ROUGE-1 e 34,24% no ROUGE-2.

5.6 Considerações Finais

Apresentou-se uma abordagem não supervisionada e baseada em otimização para resumo automático de texto. No método proposto, a sumarização de texto é formulada como um problema de otimização combinatória quadrática inteira. Os critérios a otimizar incluem a maximização da relevância das frases selecionadas, a minimização da redundância através da promoção da diversidade, a maximização da informatividade das frases selecionadas e a garantia do cumprimento de uma restrição de comprimento máximo do resumo. A abordagem proposta é aplicável a tarefas de resumo de documentos únicos e multidocumentos.

Neste artigo, um algoritmo memético, *Maximum Relevance with Minimum Redundancy using Shuffle Frog-Leaping Algorithm (MRMRSFLA)*, foi projetado, implementado e desenvolvido pela primeira vez para resolver este problema. MRMRSFLA é um algoritmo de inteligência de enxame (*swarm*) baseado em população que introduz um novo critério de otimização (informatividade), juntamente com um operador de mutação especificamente adaptado ao problema genérico de sumarização. No MRMRSFLA, a exploração das melhores soluções (busca local) é realizada dentro de *memeplexes* (soluções candidatas). Além disso, as soluções candidatas são periodicamente embaralhadas e reorganizadas em novos *memeplexes* para garantir uma busca global.

Os experimentos foram conduzidos usando os conjuntos de dados de *benchmark* DUC2001 e DUC2002. Esses conjuntos de dados são amplamente utilizados na literatura para validar a tarefa de sumarização automática de textos e são compostos por notícias e seus respectivos resumos feitos por humanos. Após realizar uma análise estatística em 835 documentos, os resultados indicam que o MRMRSFLA proporciona resultados superiores em comparação com outras abordagens na literatura científica. Um total de 16 métodos de outros autores foram

utilizados para comparação. MRMRSFLA alcançou uma melhoria percentual média de 14,55% no ROUGE-1 e 23,38% no ROUGE-2 no conjunto de dados DUC2001. Para o DUC2002, obteve uma melhoria média de 27,64% no ROUGE-1 e 34,24% no ROUGE-2. A métrica de avaliação de interesse para ambos os casos foi a *Recall*.

A seguir, no Capítulo 6, o método *Holistic Text Summarization with the Shuffled Frog-Leaping Algorithm* (HSSFLA) é apresentado, que aborda o problema de resumo extrativo automático de forma holística.

6

Resumo Extrativo Holístico

Este capítulo reproduz parcialmente o artigo *A Generic Extractive Multi-document Text Summarization Method Using Memetic Algorithm and Combinatorial Optimization*, **aceito** na XXII Simpósio Brasileiro de Sistemas de Informação (SBSI). Nele, apresentamos o método *Holistic Text Summarization with the Shuffled Frog-Leaping Algorithm* (HSSFLA) de resumo extrativo genérico multidocumento, buscando mitigar as limitações encontradas no mapeamento sistemático e na revisão de literatura sobre resumo automático por meio de métodos bioinspirados.

6.1 Principais Contribuições

Este artigo concentra-se no problema de sumarização de texto genérica, extrativa, multi-documento e multilíngue. Neste estudo, o método *Holistic Text Summarization with Shuffle Frog-Leaping Algorithm* (HSSLF) foi desenvolvido, implementado e aplicado para resolver o problema de sumarização de texto genérica, extrativa, multi-documento e multilíngue. Experimentos *in vitro* foram conduzidos utilizando os conjuntos de dados DUC 2001 e DUC 2002 (NIST, 2025). Os resultados foram avaliados por meio da métrica padrão para avaliação de resumos automáticos, *Recall-Oriented Understudy for Gisting Evaluation* (ROUGE) (LIN, 2004; EL-KASSAS et al., 2021).

As limitações discutidas na Subseção 3.4.4 e na Seção 4 também são endereçadas pelo HSSFLA, como:

- Avaliação robusta por meio da definição de seus parâmetros utilizando o conjunto de dados DUC2001 e da mensuração de seu desempenho também no DUC2002, demonstrando sua capacidade de generalização;
- O HSSFLA não depende de especialistas em nenhuma etapa do *pipeline*, é não supervisionado, foi projetado para ser genérico e é aplicável tanto a tarefas de sumarização de

documento único quanto de múltiplos documentos, independentemente de domínio ou idioma;

- Não descarta soluções candidatas que excedam um comprimento máximo predefinido; em vez disso, penaliza-as de forma suave proporcionalmente ao excesso de comprimento, garantindo que soluções potencialmente adequadas não sejam descartadas devido a violações mínimas de tamanho;
- Sua execução exige poucos recursos computacionais;
- Possui baixa complexidade;
- Independe do contexto linguístico, ou seja, textos especializados como a saúde — que possuem linguagem técnica, siglas, termos especializados —, não degradam a qualidade do resumo gerado.

Mas, diferentemente do MRMRSFLA (Capítulo 5), o HSSFLA utiliza todas as frases para criar, evoluir e avaliar os resumos candidatos. Ou seja, otimiza sua função objetivo e avalia a qualidade do resumo por meio da função *fitness* em todo o conteúdo, não em subgrupos como no MRMRSFLA. Além disso, um novo termo na função *fitness* foi adotado, utilizando uma penalidade progressiva (quadrática), permitindo uma exploração mais suave, penalizando menos quem excede menos o limiar de tamanho do resumo e penalizando mais quem excede mais.

Assim, as principais contribuições deste artigo podem ser resumidas da seguinte forma:

- O problema de sumarização de texto genérica, extrativa, multi-documento e multilíngue foi formulado como um problema de otimização combinatória quadrática inteira, envolvendo a otimização dos seguintes critérios: maximização da relevância do resumo, minimização da redundância e maximização da informatividade das sentenças selecionadas.
- O HSSFLA, um algoritmo memético baseado em inteligência de enxame, foi desenvolvido para resolver esse problema pela primeira vez.
- O HSSFLA gera resumos holísticos, nos quais a qualidade do resumo é avaliada como um todo, em vez de se concentrar exaustivamente na identificação das melhores sentenças individuais.
- Experimentos foram conduzidos utilizando os conjuntos de dados DUC2001 e DUC2002, com avaliação por meio das métricas ROUGE.
- O HSSFLA supera os resultados reportados na literatura científica nos conjuntos de dados DUC2001 e DUC2002.

O restante deste artigo está organizado da seguinte forma. A Seção 6.2 formula o problema de sumarização genérica de texto multi-documento como um problema de otimização combinatória quadrática inteira. A Seção 6.3 descreve a função de *fitness* utilizada para avaliar a qualidade dos resumos candidatos gerados pelo algoritmo HSSFLA. A Seção 6.4 introduz o algoritmo básico Shuffled Frog-Leaping Algorithm (SFLA), descreve em detalhes o Holistic Text Summarization with Shuffled Frog-Leaping Algorithm (HSSFLA) e a estratégia de mutação. A Seção 6.5 descreve as etapas de pré-processamento aplicadas aos documentos brutos antes da execução do algoritmo, apresenta os conjuntos de dados utilizados, as métricas de avaliação, as configurações de parâmetros, os resultados obtidos com a abordagem proposta e as comparações com outros métodos da literatura. Por fim, a Seção 6.6 conclui o artigo e discute direções para pesquisas futuras.

6.2 Formulação Matemática do Problema de Otimização

O problema de otimização é formalmente definido da seguinte forma. Seja $D = d_1, d_2, d_3, \ldots, d_N$ uma coleção de documentos composta por N documentos. Alternativamente, essa coleção pode ser representada como $D = s_1, s_2, s_3, \ldots, s_n$, em que n corresponde ao número total de sentenças extraídas de todos os documentos da coleção. O objetivo é gerar um resumo S por meio da seleção de um subconjunto de sentenças de D ($S \subset D$), atendendo aos seguintes critérios:

- **Maximização da relevância**: o resumo deve incluir sentenças que sejam mais relevantes de acordo com os tópicos centrais dos documentos.
- **Minimização da redundância**: o resumo deve evitar a inclusão de sentenças excessivamente semelhantes entre si.
- **Informatividade**: o resumo deve conter sentenças que forneçam o conteúdo mais informativo.
- **Restrição de comprimento**: o resumo deve possuir um comprimento predefinido L.

A maximização da relevância concentra-se na seleção de sentenças altamente representativas em relação ao centro do documento, enquanto a redução da redundância visa evitar a construção de resumos que incluam sentenças com elevada similaridade entre si. Adicionalmente, a informatividade assegura que o conjunto de sentenças selecionadas contribua coletivamente com informações diversas e relevantes. Para atender a esses requisitos, a função objetivo proposta por Alguliyev, Aliguliyev e Isazade (2015) foi adaptada e aprimorada, de modo a melhorar a seleção de sentenças informativas por meio da incorporação da maximização da entropia sobre os valores de relevância (α). Consequentemente, a seguinte função objetivo foi definida:

$$\text{maximize} \quad f(X) = \text{max\_rel} \cdot \text{min\_red} + \gamma H(X) \tag{6.1}$$

$$\text{subject to} \sum_{i=1}^{n} l_i x_i \leq L, \quad x_i \in \{0,1\}, \text{ for } i = 1, \ldots, n \tag{6.2}$$

Em que l_i representa o comprimento da sentença S_i. O número de palavras mede tanto o tamanho do resumo quanto o da sentença. Os termos que maximizam a relevância das sentenças (max_rel), minimizam a seleção de sentenças redundantes (min_red) e promovem a informatividade são formulados da seguinte forma:

$$\text{max\_rel} = \left(\sum_{i=1}^{n} \alpha_i x_i \right) \tag{6.3}$$

$$\text{min\_red} = \frac{1}{\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \text{cosim}(s_i, s_j) \cdot x_i x_j + \epsilon} \tag{6.4}$$

A relevância relativa (α) das sentenças candidatas ao resumo foi calculada da seguinte maneira:

$$\alpha_i = \frac{\text{sim}(s_i, O)}{\sum_{j=1}^{n} \text{sim}(s_j, O)}, \text{i = 1, 2, ..., n} \tag{6.5}$$

Em que O é o centro da coleção $D = \{s_1, s_2, ..., s_n\}$, e a k-ésima coordenada o_k do centro é calculada como:

$$o_k = \frac{1}{n} \sum_{i=1}^{n} w_{i,k} \tag{6.6}$$

O peso definido na Equação (6.5) determina a relevância relativa da sentença s_i em relação ao conteúdo principal da coleção D.

Na Equação 6.1, Com o objetivo de aumentar a informatividade (*informativeness*), a entropia ($\gamma H(X)$) é calculada com base no vetor de relevância (α) correspondente às sentenças selecionadas, o qual representa sua distribuição de probabilidade.

A entropia atua como uma medida da diversidade informacional no conjunto de sentenças selecionadas. Um valor de entropia mais elevado indica uma distribuição mais equilibrada e heterogênea dos escores de relevância, sugerindo que o resumo é composto por conteúdos variados e informativos, sem que uma única sentença influencie de forma desproporcional o resumo. Essa maior incerteza reflete uma cobertura mais ampla das informações relevantes (KHURANA; BHATNAGAR, 2022).

Ao integrar a entropia ao processo de seleção, o método promove uma escolha equilibrada de sentenças com diferentes níveis de relevância, melhorando, assim, a informatividade geral do resumo. A entropia é calculada pela Eq. 6.7, de acordo com Shannon (1948).

$$h = -\sum p_i \cdot \log_2(p_i) \tag{6.7}$$

6.3 Função Fitness

Uma função de *fitness* é definida para avaliar a qualidade de um resumo candidato, uma vez que seu valor atua como um indicador de quão adequadamente o resumo atende ao problema de otimização.

Um termo de penalidade foi incorporado à função de *fitness* para transformar o problema restrito em um problema irrestrito.

Um termo de penalidade adicional é introduzido, conforme Alguliyev, Aliguliyev e Isazade (2015), para desencorajar soluções inviáveis por meio da aplicação de um fator de penalidade $\beta(\beta > 0)$. Além disso, a função de *fitness* proposta utiliza uma penalidade progressiva (quadrática), permitindo uma exploração mais suave nas proximidades do limite de comprimento e penalizando severamente as soluções que o excedem de forma significativa.

A função de *fitness* é formalmente definida da seguinte forma:

$$fit(X) = f(X) \cdot \exp\left(-\beta \cdot \left(\max\left(0, \sum_{i=1}^{n} l_i \cdot x_i - L\right)\right)^2\right) \tag{6.8}$$

6.4 Holistic Text Summarization with Shuffled Frog-Leaping Algorithm

Nesta seção, apresentamos o *Holistic Text Summarization with Shuffled Frog-Leaping Algorithm* (HSSFLA). Inicialmente, descreve-se o algoritmo básico SFLA. Em seguida, são definidos os passos de pré-processamento. Por fim, são explicadas as principais etapas do HSSFLA e seus operadores centrais.

6.4.1 Algoritmo Base

A metaheurística memética Shuffled Frog-Leaping Algorithm (SFLA), introduzida por Eusuff, Lansey e Pasha (2006), foi projetada para resolver problemas de otimização combinatória. O SFLA é um método de busca cooperativa baseado em população, inspirado no conceito de memética natural. O algoritmo opera dividindo a população de soluções candidatas (sapos) em grupos denominados memeplexes, dentro dos quais os indivíduos interagem e compartilham

informações. Nesse contexto, os sapos virtuais atuam como portadores de memes, em que um meme representa uma unidade de evolução cultural. O algoritmo realiza buscas locais independentes dentro de cada memeplex, ao mesmo tempo em que executa uma exploração global por meio do embaralhamento periódico dos sapos e de sua reorganização em novos memeplexes. O procedimento detalhado do SFLA é apresentado no Algoritmo 3.

Algorithm 3 Basic SFLA pseudocode

```
Pop ← init_population(popsize)
Pop ← calculate_fitness(Pop)
Pop ← sort_by_fitness(Pop)
while not stop_criteria do
    Memplexes ← divide_pop_into_memplexes(Pop, num)
    Memplexes ← local_search(Memplexes, num, improvsmax)
    Pop ← combine_evolved_memplexes(Memplexes)
    Pop ← calculate_fitness(Pop)
    Pop ← sort_by_fitness(Pop)
end while=0
```

6.4.2 Principais etapas do Topic HSSFLA

A metaheurística SFLA foi selecionada como base para o tratamento do problema de sumarização extrativa genérica devido à baixa complexidade envolvida em sua adaptação para uma abordagem de sumarização genérica (HSSFLA). Além disso, apresenta a vantagem de realizar buscas locais e globais para otimizar a função objetivo (Equação 5.1) de maneira relativamente simples. O algoritmo HSSFLA detalhado é descrito no Algoritmo 4. Adicionalmente, o código com a implementação está disponível em um repositório público na plataforma GitHub[1].

6.4.3 Mutação

Como o método apresentado neste estudo consiste em uma abordagem de sumarização genérica, todas as comparações são realizadas com o centróide, e não com a consulta, conforme originalmente proposto em Sanchez-Gomez, Vega-Rodríguez e Pérez (2022). Esta seção descreve a estratégia de mutação proposta por adaptada para o HSSFLA.

O processo de mutação consiste na modificação de um sumário candidato (indivíduo) por meio da execução de uma dentre três operações: adição (Eq. 5.9), remoção (Eq. 5.10) ou substituição de uma sentença, de acordo com o tipo de mutação selecionado. Cada tipo de mutação possui a mesma probabilidade de ser escolhido, sendo apenas um aplicado em cada evento de mutação. A probabilidade de mutação é definida como $p_m = 1/n$, em que n denota o número total de sentenças, garantindo que exatamente uma sentença seja alterada por iteração. A mutação é aplicada de forma incondicional, independentemente da sentença modificada contribuir para a melhoria da qualidade da solução.

[1] https://github.com/k3ybladewielder/hssfla

Algorithm 4 HSSFLA pseudocode

```
1: Population ← init_population(popsize)
2: Population ← calculate_objective_functions(Population, popsize)
3: Population ← sort_by_fitness(Population, popsize)
4: for cycle = 1 to cyclesmax do
5:    X_bestG ← select_best_global(Population)
6:    Memplexes ← divide_pop_into_memplexes(Population, memnum)
7:    for m = 1 to memesnum do
8:       for i = 1 to improvsmax do
9:          X_bestL ← select_best_local(Memplexes[m])
10:         X_worstL ← select_worst_local(Memplexes[m])
11:         save_worst_local(Population, X_worstL)
12:         X_new ← mutate_solution(X_bestL, p_m)
13:         if X_new ≻ X_worstL then
14:            save_solution(Memplexes[m], X_new)
15:         else
16:            X_new ← mutate_solution(X_bestG, p_m)
17:            if X_new ≻ X_worstL then
18:               save_solution(Memplexes[m], X_new)
19:            else
20:               X_new ← random_solution()
21:               save_solution(Memplexes[m], X_new)
22:            end if
23:         end if
24:         Memplexes[m] ← sort_by_dominance(Memplexes[m])
25:      end for
26:   end for
27:   Population ← combine_evolved_memplexes(Memplexes)
28:   Population ← calculate_objective_functions(Population, popsize * 2)
29:   Population ← sort_by_fitness(Population, popsize * 2)
30:   Savebest_individual, best_fitness, best_sentences
31: end for=0
```

O operador de adição envolve a incorporação de uma sentença do conjunto de sentenças disponíveis que ainda não faz parte do sumário candidato. Espera-se que a sentença selecionada ($s_i \notin S$) melhore a qualidade do sumário. Especificamente, a similaridade do cosseno entre a nova sentença e o centróide deve ser superior à similaridade média do cosseno das sentenças existentes no sumário candidato em relação ao centróide:

$$cosim(s_i, O) > \frac{1}{n}\sum_{j=1}^{n} cosim(s_j, O). \tag{6.9}$$

Uma sentença $s_i \notin S$ é selecionada aleatoriamente do conjunto de documentos D e adicionada ao sumário candidato caso satisfaça a condição especificada. Caso a condição não seja atendida, a sentença $s_i \notin S$ com a maior similaridade do cosseno em relação ao centróide é selecionada e adicionada ao sumário.

O operador de remoção de sentença elimina uma sentença do sumário candidato. Para

garantir que a remoção não afete negativamente a qualidade do sumário, a similaridade do cosseno entre a sentença $s_i \in S$ e o centróide deve ser inferior à similaridade média do cosseno de todas as sentenças atualmente presentes no sumário em relação ao centróide:

$$cosim(s_i, O) < \frac{1}{n} \sum_{j=1}^{n} cosim(s_j, O). \tag{6.10}$$

De forma semelhante ao processo de adição, uma sentença $s_i \in S$ é selecionada aleatoriamente do sumário candidato. Caso satisfaça a condição especificada, ela é removida; caso contrário, sentenças subsequentes $s_i \in S$ são avaliadas até que a condição seja atendida. Se nenhuma sentença satisfizer o critério de remoção, a sentença $s_i \in S$ com a menor similaridade do cosseno em relação ao centróide é removida.

A operação de substituição de sentença consiste na troca de uma sentença da coleção de documentos D que não esteja atualmente incluída no sumário por uma sentença que já faça parte dele. Esse processo de mutação remove simultaneamente uma sentença do sumário e incorpora uma sentença distinta proveniente da coleção de documentos.

6.5 Resultados experimentais

Esta seção descreve as etapas de pré-processamento, os conjuntos de dados DUC2001 e DUC2002 utilizados, a métrica de avaliação de desempenho, o processo de definição dos parâmetros, os resultados obtidos com a abordagem proposta e a comparação com resultados da literatura.

6.5.1 Pré-processamento

Antes da execução do algoritmo, algumas etapas de pré-processamento devem ser realizadas nos documentos da coleção D. Essas etapas são descritas a seguir:

1. **Segmentação**: cada sentença do documento é extraída individualmente, com a definição clara de seu início e fim.

2. **Tokenização**: todas as palavras de uma sentença são tokenizadas com o objetivo de remover caracteres especiais, sinais de pontuação e símbolos como pontos de interrogação.

3. **Remoção de stopwords**: palavras que não carregam significado semântico relevante e que aparecem com alta frequência, tais como preposições, conjunções e artigos, são removidas das sentenças. A operação de remoção de stopwords foi realizada utilizando a biblioteca NLTK[2].

[2]https://www.nltk.org/

4. ***Stemming***: o radical de cada palavra foi extraído por meio do módulo SnowballStemmer da biblioteca NLTK.

5. **Representação**: as palavras das sentenças são representadas numericamente por meio da aplicação de TF-ISF.

6.5.2 Base de Dados

O desempenho do algoritmo proposto foi avaliado utilizando os conjuntos de dados DUC2001 e DUC2002, que são corpora de referência de acesso público disponibilizados pela *Document Understanding Conference* (DUC) (NIST, 2025) para a avaliação de sistemas automáticos de sumarização de textos. Esses conjuntos de dados são compostos por artigos jornalísticos em língua inglesa, abrangendo uma ampla variedade de tópicos, provenientes de fontes como *Financial Times*, *Associated Press* e *The Wall Street Journal*. Especificamente, o DUC2001 inclui 30 tópicos com um total de 302 documentos, enquanto o DUC2002 contém 59 tópicos que totalizam 533 documentos. Ambos os conjuntos de dados disponibilizam sumários de referência, cada um limitado a 100 palavras, elaborados manualmente por especialistas humanos para cada documento. Cada instância dos conjuntos de dados consiste no documento original acompanhado de seu respectivo sumário produzido por humanos, o qual é utilizado como padrão-ouro para a avaliação. Uma descrição detalhada dos conjuntos de dados DUC2001 e DUC2002 é apresentada na Tabela 24.

Tabela 24 – Descrição dos conjuntos de dados DUC2001 e DUC2002.

	DUC2001	DUC2002
No. de Tópicos	30	59
No. de Documentos	302	533
No. médio de Sentenças por Tópico	59.5	150.55
No. médio de Sentenças por Documento	5.91	15.55
No. médio de Termos por Tópico	1012.97	2138.68
No. médio de Termos por Documento	100.62	236.74

Em média, os sumários de referência do conjunto de dados DUC2002 são mais longos, tanto em termos do número de sentenças por tópico e por documento quanto do número de palavras por tópico e por documento. Todos os documentos foram segmentados em sentenças e submetidos ao pré-processamento. O processo de pré-processamento dos dados é descrito na Seção 6.5.1.

6.5.3 Definição de Parâmetros

Para definir os parâmetros, foi realizado um teste utilizando uma amostra do conjunto DUC2001. A amostra foi determinada com nível de confiança de 0,95 no cálculo do tamanho

amostral, resultando em 169 (55%) exemplos. O teste consistiu na execução do algoritmo com 5, 10 e 15 ciclos (*cyclesmax*), selecionando aleatoriamente valores para o número de iterações (*improvsmax*), o número de *memeplexes* (*memesnum*), o parâmetro gamma (γ) e o tamanho da população (*pop_size*), avaliando seu desempenho por meio das métricas descritas na Subseção 2.7.1. Os valores possíveis para o número de iterações (*improvsmax*) foram 25, 50, 75 e 100; para o número de *memeplexes* (*memesnum*), 5, 10, 15, 20 e 25; para o parâmetro gamma (γ), 0.25, 0.5, 0.75 e 0.9; enquanto os valores possíveis para o tamanho da população (*pop_size*) foram 100, 200, 300, 400 e 500. O teste foi realizado de forma independente 31 vezes para cada valor do parâmetro de ciclos (*cyclesmax*) e, em cada execução, os parâmetros foram selecionados aleatoriamente, todos com igual probabilidade de escolha.

Após a realização de 31 execuções para cada documento da amostra calculada, foram geradas, de forma aleatória, um total de 92 configurações de parâmetros. Para identificar os melhores parâmetros, as 20 configurações com maior média de ROUGE-1 foram selecionadas e armazenadas. Em seguida, também foram selecionadas as 20 configurações com maior média da medida F de ROUGE-1. Ambos os conjuntos foram combinados em uma única lista, removendo-se as configurações duplicadas, resultando em 20 configurações únicas.

Posteriormente, foi aplicado um teste de hipótese ANOVA de uma via utilizando a biblioteca *Scipy*[3] com o objetivo de verificar se alguma configuração apresentava média significativamente superior às demais. O teste ANOVA de uma via avalia a hipótese nula de que dois ou mais grupos compartilham a mesma média populacional. Quando aplicado ao conjunto completo de configurações de parâmetros, utilizando ROUGE-1 para formar os grupos, o teste resultou em uma estatística F de 0.35645 e um valor de p de 0.99999. Quando aplicado às 20 melhores configurações, os resultados foram uma estatística F de 0.2414 e um valor de p de 0.9996.

Os baixos valores da estatística F em ambos os casos indicam que a variabilidade entre as médias dos grupos é muito pequena em comparação com a variabilidade dentro de cada grupo. Em outras palavras, as médias entre os grupos são altamente semelhantes. Em ambos os cenários, considerando um nível de significância α igual a 0.05, os elevados p-valores ($p > \alpha$) indicam que a probabilidade de não se observar diferenças nos dados amostrais é substancialmente alta, assumindo que a hipótese nula seja verdadeira. Consequentemente, não há evidências estatísticas suficientes para rejeitar a hipótese nula de que a média amostral de qualquer configuração (seja considerando todas as configurações ou apenas as 20 melhores) apresente diferença estatisticamente significativa.

Dessa forma, para selecionar a configuração de parâmetros utilizada nos experimentos com os conjuntos de dados DUC2001 e DUC2002, foi realizada uma avaliação ponderada das métricas ROUGE. Os pesos atribuídos foram 0.4 para o *Recall* e para a medida F de ROUGE-1, e 0.1 para a medida F de ROUGE-2. Cada métrica foi multiplicada por seu respectivo

[3] https://scipy.org/

peso, e os valores resultantes foram somados para o cálculo de uma pontuação ponderada. A Tabela 25 apresenta os resultados para cada configuração, destacando aquela que obteve a maior pontuação ponderada. Na coluna **Config**, os valores correspondem, respectivamente, aos parâmetros *cyclesmax*, *pop_size*, *improvsmax*, *memesnum* e *gamma*.

Tabela 25 – Os 20 melhores resultados obtidos no DUC2001 com diferentes configurações e suas métricas de avaliação ROUGE.

Config	ROUGE-1 R	ROUGE-2 R	ROUGE-1 F1	ROUGE-2 F1	Weighted
10-200-25-5-0.5	0.895009	0.941808	0.933316	0.934096	0.918920
5-200-50-20-0.5	0.836247	0.917020	0.881764	1.000000	0.878906
15-300-100-5-0.9	1.000000	1.000000	0.728359	0.816262	0.872970
5-400-100-25-0.5	0.828825	0.957269	0.771117	0.951900	0.830894
10-300-25-5-0.9	0.634915	0.505620	1.000000	0.823915	0.786919
10-200-25-15-0.9	0.776983	0.956226	0.655743	0.949577	0.763670
15-200-25-15-0.5	0.820917	0.841579	0.666224	0.827401	0.761755
15-400-100-10-0.9	0.704377	0.828306	0.735278	0.831737	0.741866
10-500-25-15-0.25	0.901444	0.901595	0.554950	0.647784	0.737495
10-500-25-10-0.25	0.764803	0.680823	0.726792	0.658799	0.730600
5-200-50-10-0.9	0.660885	0.694636	0.787702	0.746091	0.723507
5-200-100-5-0.75	0.751163	0.774060	0.674126	0.729329	0.720455
5-400-100-5-0.5	0.638754	0.728994	0.763853	0.815231	0.715466
5-100-75-20-0.25	0.556204	0.824279	0.693786	0.960909	0.678515
10-100-50-10-0.5	0.727246	0.673544	0.652443	0.568588	0.676088
5-300-75-5-0.25	0.592966	0.525563	0.754388	0.743605	0.665858
10-300-75-5-0.75	0.569153	0.695551	0.701375	0.811051	0.658871
15-500-75-10-0.25	0.681431	0.868207	0.508022	0.843672	0.646969
15-100-25-5-0.75	0.701763	0.801257	0.540416	0.668539	0.643851
15-300-25-10-0.75	0.710391	0.708510	0.575787	0.561802	0.641503

6.5.4 Resultados com o Método Proposto

Nesta subseção, são apresentados os resultados obtidos com a execução do HSSFLA nas bases de dados DUC2001 e DUC2002. Foi realizada uma análise estatística dos escores ROUGE-1 e ROUGE-2 para todos os tópicos ao longo dos conjuntos de dados utilizados.

As tabelas incluem a média, a mediana e o desvio padrão, bem como o primeiro e o terceiro quartis (Q1 e Q3), além dos valores mínimo e máximo para os escores ROUGE. Esses valores foram calculados com base em 31 repetições independentes por amostra, tanto para o conjunto DUC2001 quanto para o DUC2002. As Tabelas 26 e 27 apresentam os resultados obtidos.

A média de ROUGE-N representa o valor esperado ou o desempenho médio do modelo ao longo de todas as amostras avaliadas. A mediana indica o valor central da distribuição, ou seja, 50% dos resumos obtiveram desempenho acima desse valor e 50% abaixo. A mediana é menos sensível a valores extremos (*outliers*) do que a média, oferecendo, assim, uma medida mais robusta do desempenho típico. O primeiro quartil (Q1) representa o valor abaixo do qual

Tabela 26 – Resultados obtidos pelo HSSFLA para ROUGE-1 e ROUGE-2 utilizando o DUC2001

HSSFLA	ROUGE-1 Precision	ROUGE-2 Precision	ROUGE-1 Recall	ROUGE-2 Recall	ROUGE-1 F1	ROUGE-2 F1
Média	0.187162	0.082696	0.594694	0.265815	0.257536	0.116835
Mediana	0.168359	0.064916	0.626866	0.253521	0.251121	0.100840
Desvio Padrão	0.102535	0.077730	0.194115	0.153408	0.109598	0.083058
Q1	0.119116	0.036649	0.514706	0.161765	0.186880	0.059406
Q3	0.232759	0.107289	0.722222	0.354839	0.325792	0.157521
Mínimo	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
Máximo	1.000000	1.000000	1.000000	0.956522	0.769231	0.695652

Tabela 27 – Resultados obtidos pelo HSSFLA para ROUGE-1 e ROUGE-2 utilizando o DUC2002

HSSFLA	ROUGE-1 Precision	ROUGE-2 Precision	ROUGE-1 Recall	ROUGE-2 Recall	ROUGE-1 F1	ROUGE-2 F1
Média	0.236632	0.111303	0.575184	0.263056	0.304667	0.140688
Mediana	0.209302	0.087591	0.600000	0.250000	0.299270	0.126214
Desvio Padrão	0.128821	0.107376	0.188940	0.143518	0.112938	0.086880
Q1	0.156425	0.053435	0.485294	0.163636	0.234604	0.080460
Q3	0.287770	0.138554	0.701493	0.352941	0.375000	0.186916
Mínimo	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
Máximo	1.000000	1.000000	1.000000	0.936508	0.857143	0.725664

se encontram 25% dos resultados. No contexto do ROUGE, Q1 reflete o desempenho nos 25% piores casos, auxiliando na avaliação da consistência e da estabilidade do método em cenários menos favoráveis. O terceiro quartil (Q3) representa o valor abaixo do qual se encontram 75% dos resultados, abrangendo, portanto, a maior parte dos casos, excetuando-se os 25% superiores. O Q3 contribui para a compreensão da variabilidade do desempenho, indicando o quão bem o método se comporta para a maioria dos casos avaliados.

A média da *Precision* de ROUGE-1 aumentou de 0.187 no DUC2001 para 0.237 no DUC2002, enquanto a *Precision* de ROUGE-2 passou de 0.083 para 0.111. Isso sugere que o modelo foi capaz de gerar resumos com maior acurácia, isto é, com uma maior proporção de n-gramas gerados que efetivamente aparecem nos resumos de referência do conjunto DUC2002. Os valores de mediana e quartis confirmam essa tendência, apresentando aumentos consistentes nesses parâmetros, o que indica melhorias tanto no desempenho típico quanto na estabilidade dos resultados.

O *Recall* médio para ROUGE-1 apresentou uma leve redução, de 0.595 para 0.575, enquanto o *Recall* de ROUGE-2 permaneceu praticamente inalterado (de 0.266 para 0.263). De forma semelhante, as medianas e os quartis para *Recall* exibem apenas variações discretas, sugerindo que o modelo recupera uma proporção semelhante de n-gramas relevantes em ambos os conjuntos de dados. A ligeira diminuição do *Recall* médio de ROUGE-1 pode indicar que, apesar de alcançar maior precisão, o método recuperou uma fração menor do conteúdo relevante, possivelmente refletindo uma estratégia de seleção de sentenças mais conservadora.

O F1-score, que combina *Precision* e Recall, apresenta melhorias médias de 0.258 para 0.305 em ROUGE-1 e de 0.117 para 0.141 em ROUGE-2 ao se passar do DUC2001 para o DUC2002. Esse aumento indica um equilíbrio mais favorável entre *Precision* e *Recall* no segundo conjunto de dados, o que é corroborado pelos valores mais elevados de mediana e quartis.

Os valores do primeiro e do terceiro quartis para *Precision* e *F1-score* também são

superiores no DUC2002, indicando que tanto os 25% piores quanto os 25% melhores resultados apresentaram melhoria em comparação ao DUC2001. No entanto, o *Recall* apresenta uma leve redução em Q1 para ROUGE-1, sugerindo maior variabilidade nos casos de menor desempenho.

As Figuras 11 e 12 apresentam *boxplots* e histogramas dos resultados obtidos para ROUGE-1 e ROUGE-2.

Em síntese, a análise das estatísticas descritivas evidencia que o método HSSFLA apresenta desempenho ligeiramente superior em termos de *Precision* e *F1-score* no conjunto DUC2002 em comparação ao DUC2001. Isso indica uma maior capacidade do método em produzir resumos mais aderentes aos documentos de referência, mesmo com os parâmetros tendo sido otimizados exclusivamente a partir de amostras do DUC2001. Por outro lado, o *Recall* mantém-se relativamente estável ou apresenta leve redução, sugerindo uma possível priorização da precisão em detrimento da recuperação completa do conteúdo relevante. A estabilidade dos valores de quartis reforça a consistência desse comportamento em diferentes níveis de desempenho, destacando a robustez do método no cenário do DUC2002.

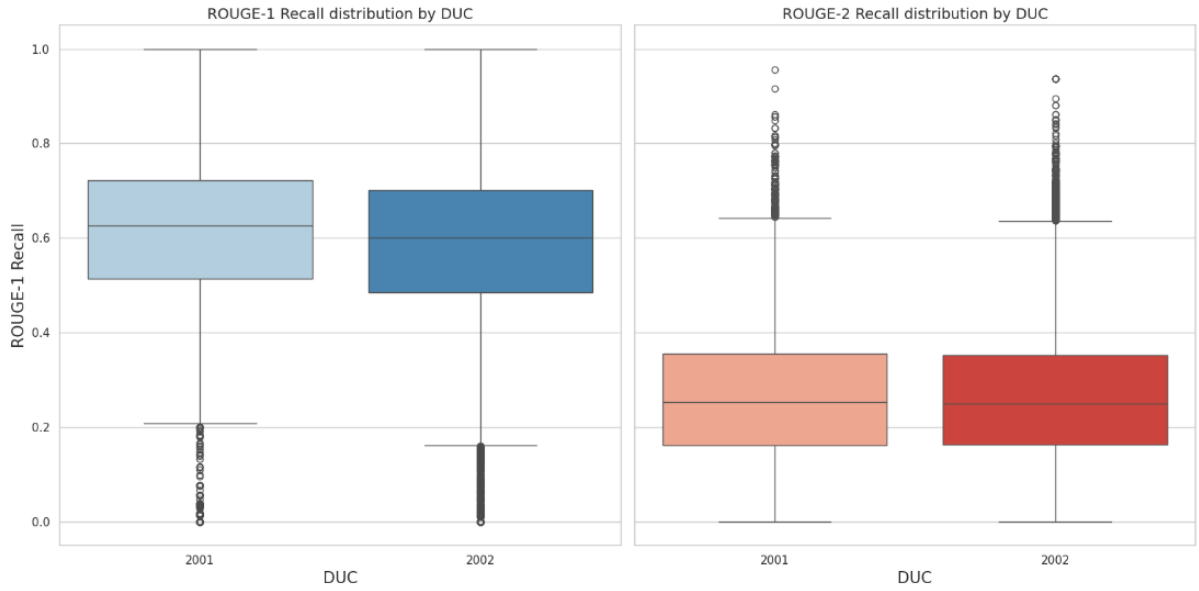


Figura 11 – *Boxplots* obtidos pelo HSSFLA para ROUGE-1 e ROUGE-2 (valores de *Recall*).

6.5.5 Comparação com resultados de outras abordagens

A subseção a seguir apresenta os resultados obtidos por outras abordagens, os quais serão comparados com os resultados da metodologia proposta neste estudo.

As Tabelas 28 e 29 apresentam os resultados comparativos para os conjuntos de dados DUC2001 e DUC2002, com base nas métricas avaliadas para cada método concorrente. Essas tabelas reportam o valor médio de *Recall* para ROUGE-1 e ROUGE-2, bem como, entre parênteses, a porcentagem de melhoria alcançada pelo HSSFLA. O símbolo de hífen é utilizado

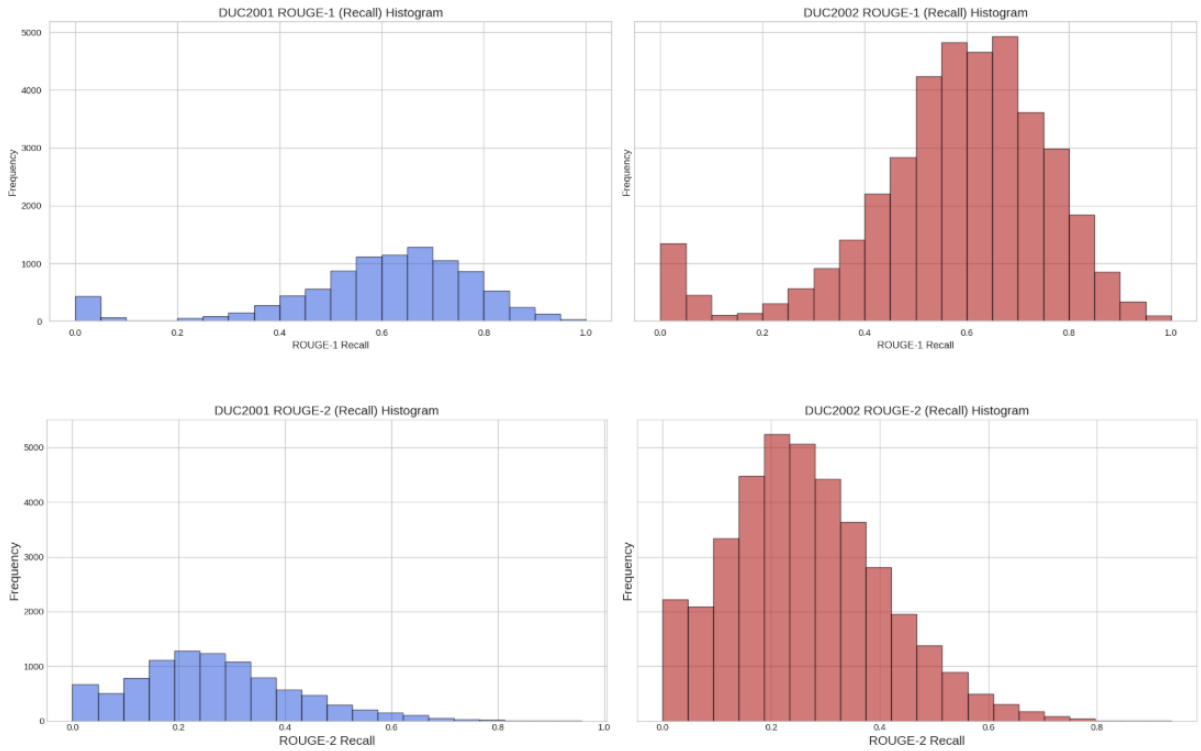


Figura 12 – Histogramas obtidos pelo HSSFLA para ROUGE-1 e ROUGE-2 (valores de *Recall*).

quando o resultado para uma determinada métrica não está disponível. A porcentagem é calculada como $((\text{métrica proposta} - \text{métrica de referência})/\text{métrica de referência}) * 100$.

Tabela 28 – Comparação do modelo proposto com outros métodos usando ROUGE-1 e ROUGE-2 no DUC2001 (Valores de *Recall*). O HSSFLA e a melhor métrica estão destacados em negrito.

Métodos	**ROUGE-1**		**ROUGE-2**	
HSSFLA	**0.594694**	-	**0.265815**	-
FEC_B&B	**0.4971**	(+19.63)	**0.2112**	(+25.86)
FEC_Gap	0.4865	(+22.24)	0.2081	(+27.73)
FEC_GAP*	0.4816	(+23.48)	0.2054	(+29.41)
FEC without Gap	0.4632	(+28.39)	0.1873	(+41.92)
FEC_WGAP	0.4621	(+28.69)	0.1866	(+42.45)
FEC_WGAP*	0.4613	(+28.92)	0.1831	(+45.17)
Average Result	0.4754	(+25.12)	0.1971	(+34.91)

Para o DUC2001, os métodos de estado da arte utilizados para comparação incluem *Branch and Bound* (B&B), estatísticas de *gap* padrão (Gap), estatísticas de *gap* sem a função logarítmica (Gap*), estatísticas de *gap* ponderadas (WGap) e estatísticas de *gap* ponderadas sem a função logarítmica (WGap*), conforme proposto por Verma, Verma e Pal (2022), os quais foram comparados com o HSSFLA.

Para o DUC2002, as comparações foram realizadas em relação aos métodos FbTS (TOMER; KUMAR, 2022), FUZZY CST com COM (KUMAR et al., 2014), FEC sem Gap

Tabela 29 – Comparação do modelo proposto com outros métodos usando ROUGE-1 e ROUGE-2 no DUC2002 (Valores de *Recall*). O HSSFLA e a melhor métrica estão destacados em negrito.

Métodos	**ROUGE-1**		**ROUGE-2**	
HSSFLA	**0.575184**	-	**0.263056**	-
FEC_B&B	**0.4987**	(+15.34)	**0.2187**	(+20.28)
FEC without Gap	0.4637	(+24.04)	0.1889	(+39.26)
FEC_WGAP	0.4636	(+24.07)	0.2138	(+23.04)
FEC_GAP*	0.4634	(+24.12)	0.1966	(+33.80)
FEC_Gap	0.4613	(+24.69)	0.2163	(+21.62)
FEC_WGAP*	0.4613	(+24.69)	0.1882	(+39.77)
Punctuation+Root Stemming	0.4572	(+25.81)	-	-
FbTS	0.4380	(+31.31)	0.2121	(+24.01)
Punctuation+Lemma Stemming	0.352	(+63.40)	-	-
FUZZY CST with COM	0.33206	(+73.22)	0.12806	(+105.42)
Average Result	0.4391	(+35.42)	0.1953	(+36.08)

(VERMA; VERMA; PAL, 2022), FEC_B&B (VERMA; VERMA; PAL, 2022), FEC_Gap (VERMA; VERMA; PAL, 2022), FEC_GAP* (VERMA; VERMA; PAL, 2022), FEC_WGAP (VERMA; VERMA; PAL, 2022), FEC_WGAP* (VERMA; VERMA; PAL, 2022), Punctuation+Lemma Stemming (ALQAISI; GHANEM; QAROUSH, 2020) e Punctuation+Root Stemming (ALQAISI; GHANEM; QAROUSH, 2020).

A abordagem proposta, HSSFLA, superou os métodos de estado da arte em ambos os benchmarks DUC2001 e DUC2002. Para o DUC2001, alcançou um aumento de 19,63% e 25,86% no *Recall* médio para ROUGE-1 e ROUGE-2, respectivamente, em comparação com o melhor método existente. Quando comparado ao método com pior desempenho, o ganho foi ainda mais expressivo, atingindo 28,92% para ROUGE-1 e 45,17% para ROUGE-2.

No DUC2002, o HSSFLA obteve um ganho de 15,34% e 20,28% em ROUGE-1 e ROUGE-2, respectivamente, em relação ao melhor método concorrente. Em comparação com os métodos de menor desempenho, a melhoria foi ainda mais substancial, alcançando 73,22% para ROUGE-1 e expressivos 105,42% para ROUGE-2.

Em média, a abordagem proposta resultou em um aumento percentual de 25,12% em ROUGE-1 e 34,91% em ROUGE-2 para o conjunto de dados DUC2001. Por sua vez, para o DUC2002, foram observados ganhos médios percentuais de 35,42% em ROUGE-1 e 36,08% em ROUGE-2.

6.6 Considerações Finais

Apresenta-se uma abordagem não supervisionada, baseada em otimização, para sumarização automática de textos. No método proposto, a sumarização de textos é formulada como um problema de otimização combinatória quadrática inteira. Os critérios a serem otimizados incluem a maximização da relevância do resumo candidato selecionado, a minimização da

redundância por meio da promoção da diversidade, a maximização da informatividade das sentenças selecionadas e o atendimento à restrição de comprimento máximo do resumo. A abordagem proposta é aplicável tanto a tarefas de sumarização de documento único quanto de múltiplos documentos.

Neste artigo, um algoritmo memético, denominado Holistic Text Summarization with Shuffle Frog-Leaping Algorithm (HSSFLA), foi projetado, implementado e desenvolvido pela primeira vez para resolver esse problema. O HSSFLA é um algoritmo de inteligência de enxame baseado em população que introduz um novo critério de otimização (informatividade), juntamente com um operador de mutação especificamente adaptado ao problema genérico de sumarização. No HSSFLA, a exploração das melhores soluções (busca local) é realizada dentro de *memeplexes* (soluções candidatas). Adicionalmente, as soluções candidatas são periodicamente embaralhadas e reorganizadas em novos memeplexes, de modo a garantir uma busca global.

Os experimentos foram conduzidos utilizando os conjuntos de dados de referência DUC2001 e DUC2002. Após a realização de uma análise estatística em 835 documentos, os resultados indicam que o HSSFLA apresenta desempenho superior em comparação com outras abordagens bioinspiradas descritas na literatura científica. Um total de 16 métodos propostos por outros autores foi utilizado para fins de comparação. O HSSFLA obteve um aumento percentual médio de 25,12% em ROUGE-1 e 34,918% em ROUGE-2 no conjunto de dados DUC2001. Por sua vez, no DUC2002, alcançou um ganho médio de 35,424% em ROUGE-1 e 36,08% em ROUGE-2. Em ambos os casos, a métrica de avaliação de interesse foi o recall.

No Capítulo a seguir, métodos de resumo abstrativo baseados em modelos de linguagem são explorados por meio de uma avaliação experimental.

7

Avaliação Experimental de Métodos de Resumo de Texto

Diante do gap de pesquisa existente sobre a utilização de métodos abstrativos na saúde, conforme apresentado no Capítulo 3 e buscando expandir a base de conhecimento experimental da literatura, este capítulo apresenta parcialmente o experimento controlado apresentado no artigo *Small Language Models Applied in Text Summarization Task of Health-Related News to Improve Public Health Audit: An Experimental Case Study* **publicado** no periódico *Frontiers in Artificial Intelligence*.

7.1 Contextualização

Este estudo tem como objetivo avaliar métodos automáticos de sumarização de textos por meio da comparação da qualidade de resumos gerados por máquinas com aqueles produzidos por humanos, sob a perspectiva de Cientistas de Dados e Auditores do SUS, no contexto de auditorias realizadas pelo Departamento Nacional de Auditoria do Sistema Único de Saúde (Sistema Único de Saúde — SUS) (AudSUS).

Neste estudo, conduzimos um experimento controlado para avaliar o desempenho de Small Language Models (SLMs) em tarefas de sumarização, utilizando as métricas ROUGE-N, ROUGE-L, BLEU, METEOR e BERTScore. Adicionalmente, avaliou-se a consistência dos resultados ao longo de 35 execuções, a contribuição para a redução da sobrecarga informacional e os desempenhos pareados entre os modelos.

Os modelos NousResearch/Hermes-3-Llama-3.2-3B, Qwen/Qwen2.5-7B-Instruct e meta-llama/Llama-3.2-3B-Instruct alcançaram os maiores desempenhos médios em todas as métricas, destacando-se pela capacidade de preservar o significado contextual e sintetizar informações essenciais de forma mais eficaz do que os resumos gerados por humanos.

Os achados evidenciam o potencial dos SLMs como ferramentas para a redução da sobrecarga informacional, contribuindo para o aumento da efetividade da fase analítica das

auditorias e possibilitando uma preparação mais ágil das equipes para a etapa operacional.

7.2 Materiais e Métodos

Este é um estudo experimental, seguindo os passos apresentados por Colaço JÚNIOR (2025) para avaliação dos resultados de métodos de sumarização de texto aplicados em notícias relacionadas à saúde com indícios de irregularidade, avaliando a qualidade dos resumos utilizando as métricas ROUGE-N, ROUGE-L, BLEU, METEOR e BERTScore. O processo experimental foi descrito na Seção 1.6.

A descrição da base de dados utilizada, o processo de seleção de notícias com indícios de irregularidade e a métricas de avaliação dos resumos automáticos são descritos na Subsessão 2.9.1.

7.3 Definição Experimental

Nesta seção, são apresentados o objetivo da avaliação experimental, o planejamento, as perguntas de pesquisa, as variáveis independentes, as variáveis dependentes e as hipóteses.

7.3.1 Objetivo

Para formalizar o objetivo deste estudo, o modelo de *Goal Question Metric* (GQM) de (BASILI; WEISS, 1984) foi utilizado. Este trabalho tem como objetivo **analisar** métodos de sumarização automática de texto, **com a finalidade de** avaliar a qualidade dos resumos automáticos, **contra** resumos gerados por humanos, **com respeito às** métricas de ROUGE-1, ROUGE-2, ROUGE-L, BLEU, METEOR e BERTScore, **sob o ponto de vista** de Cientistas de Dados e Auditores do SUS, **no contexto de** auditorias na saúde pública do Brasil, realizadas pelo Departamento Nacional de Auditoria do SUS (AudSUS).

7.3.2 Planejamento

O experimento envolveu a geração e avaliação dos resumos automáticos, análise e apresentação dos resultados.

A etapa de geração de resumos automáticos consiste na aplicação de métodos de resumo de texto na base de dados de notícias com indício de irregularidade. A base de dados utilizada é descrita na Subseção 2.9.1.

Na avaliação, foram aplicadas as métricas de mensuração de qualidade de resumos ROUGE-N (LIN, 2004), ROUGE-L (LIN, 2004), BLEU (PAPINENI et al., 2002), METEOR (LAVIE; AGARWAL, 2007) e BERTScorcore (ZHANG et al., 2020). As métricas são descritas na Subseção 2.7.1.

Tabela 30 – Modelos utilizados, suas características e finalidades. Ordenado em ordem alfabética por Model Name

Modelo	**Tarefa**	**Idioma**	**Resumo**	**Categoria**
google/gemma-2b-it	Text Generation	English	Abstractive	Fine-tuned
KLSum	N/A	Multilanguage	Extractive	Graph-based
LexRank	N/A	Multilanguage	Extractive	Graph-based
LSA	N/A	Multilanguage	Extractive	Math/Statistics
maritaca-ai/sabia-7b	Text Generation	Portuguese (BR)	Abstractive	Pre-trained
meta-llama/Llama-3.2-3B-Instruct	Text Generation	Multilanguage	Abstractive	Fine-tuned
nicholasKluge/TeenyTinyLlama-460m-Chat	Text Generation	Portuguese (BR)	Abstractive	Pre-trained
NousResearch/Hermes-3-Llama-3.2-3B	Text Generation	English	Abstractive	Fine-tuned
Qwen/Qwen2.5-0.5B-Instruct	Text Generation	English	Abstractive	Fine-tuned
Qwen/Qwen2.5-1.5B-Instruct	Text Generation	English	Abstractive	Fine-tuned
Qwen/Qwen2.5-7B-Instruct	Text Generation	English	Abstractive	Fine-tuned
SumBAsic	N/A	Multilíngue	Extractive	Math/Statistics
TextRank	N/A	Multilíngue	Extractive	Graph-based
TucanoBR/Tucano-1b1-Instruct	Text Generation	Portuguese (BR)	Abstractive	Fine-tuned
TucanoBR/Tucano-2b4-Instruct	Text Generation	Portuguese (BR)	Abstractive	Fine-tuned

Na etapa seguinte, foram feitas análises sobre a média das métricas para cada método aplicado e teste de hipótese para avaliar a diferença entre os resultados. A fim de identificar se existia uma diferença significativa do resultado médio entre os métodos avaliados, aplicou-se o teste de hipótese One-way ANOVA para identificação de resultados estatisticamente diferentes entre o grupo.

Após identificar os melhores métodos, eles foram descritos comparativamente quanto à distribuição dos resultados e à consistência interna por meio da análise descritiva do desvio padrão das métricas de avaliação. Por fim, as ameaças à validade são apresentadas.

7.3.3 Seleção de Contexto

Apesar dos avanços tecnológicos, diversos processos no setor público ainda dependem de buscas manuais para a construção de conhecimento. Esse cenário também se verifica no Departamento Nacional de Auditoria do Sistema Único de Saúde (AudSUS), responsável pelo controle e fiscalização do SUS. As atividades de auditoria conduzidas pelo órgão desempenham papel essencial na gestão e utilização adequada dos recursos públicos; contudo, o processo é altamente intensivo em recursos humanos devido à elevada demanda, uma vez que os auditores devem fiscalizar todas as áreas do SUS e, adicionalmente, atender às demandas internas e externas do Ministério da Saúde (FONTES et al., 2023). Nesse contexto, o experimento proposto busca apoiar a fase analítica do processo de auditoria, etapa responsável pelo planejamento e pela preparação da equipe para a fase operativa, por meio do levantamento de informações relacionadas ao objetivo da auditoria.

7.3.4 Questões de Pesquisa

Para guiar o experimento e com o para de cumprir o objetivo do experimento, foram elaboradas as seguintes questões de pesquisa:

- RQ1: Um método de sumarização automática de texto pode apoiar o processo de auditoria ao reduzir a sobrecarga de informação sobre indícios de irregularidade ?
- RQ2: Entre os métodos de sumarização selecionados, quais são os três melhores em termos de qualidade do resumo ?

Para responder às questões de pesquisa, as seguintes hipóteses teóricas descritas na Tabela 31 foram criadas.

Tabela 31 – Questões de pesquisa e hipóteses associadas

RQ	**Hipótese nula (H_0)**	**Hipótese alternativa (H_1)**
RQ1	Os métodos de resumo automático não se equiparam ao desempenho humano.	Os métodos de resumo automático se equiparam ao desempenho humano.
RQ2	Os métodos de resumo automático avaliados não apresentam diferenças estatisticamente significativas.	Os métodos de resumo automático avaliados apresentam diferenças estatisticamente significativas.

7.3.5 Variáveis dependentes

As variáveis dependentes ou variáveis de saída foram os resumos automáticos gerados pelos modelos, dos quais podem ser derivadas as métricas de avaliação de qualidade do resumo ROUGE-N, ROUGE-L, BLEU, METEOR e BERTScore.

7.3.6 Variáveis Independentes

Neste experimento, a variável independente ou variável de entrada, é a base de dados de referência que foi elaborada para ser utilizada na avaliação dos resumos automáticos, e os modelos testados: modelos abstrativos BART (LEWIS et al., 2019), Gemma (Gemma Team et al., 2024), Sabiá (PIRES et al., 2023), Llama (Meta Team, 2024), TeenyTinyLlama (CORRêA et al., 2024), Hermes (TEKNIUM; QUESNELLE; GUANG, 2024), Qwen (Qwen Team, 2024) e Tucano (CORRÊA et al., 2024b), e os modelos extrativos TextRank (NENKOVA; MCKEOWN, 2011), LexRank (ERKAN; RADEV, 2004), LSA (STEINBERGER; JEžEK, 2004), KLSum (HAGHIGHI; VANDERWENDE, 2009) e SumBAsic (WOODSEND; LAPATA, 2011). Estes modelos foram selecionados conforme a classificação de (WANG et al., 2025), ou seja. modelos até 10 bilhões de parâmetros.

7.3.7 Seleção de Objetos

Seguindo o contexto descrito em 7.3.3, os objetos deste experimento são notícias relacionadas à área da saúde com indícios de irregularidades, conforme descrito na Seção 2.9.1. O conjunto de dados contém 421 notícias e resumos de referência gerados por humanos.

Para generalizar os resultados deste experimento para a população mais ampla de notícias, é necessário avaliar os resultados utilizando uma amostra representativa (Colaço JÚNIOR, 2025). Para o cálculo do tamanho da amostra, considerou-se uma população finita de 154.407 notícias (o número total de artigos no conjunto de dados completo). Ressalta-se que a amostra final excede o número estimado de acordo com a Eq. 7.2. No cálculo do tamanho da amostra, adotaram-se um nível de confiança de 95% (Z = 1.96), um erro amostral tolerável de 5% ($e = 0.05$) e uma proporção esperada de 50% ($p = 0.5$), maximizando a variabilidade amostral e assegurando um tamanho de amostra mais conservador.

O cálculo do tamanho da amostra para uma população finita foi realizado em duas etapas: inicialmente, estimou-se a amostra para uma população infinita (n) por meio da Eq. 7.1 e, em seguida, aplicou-se o ajuste para população finita ($n_{adjusted}$) conforme a Eq. 7.2, resultando em aproximadamente 383,21 amostras, conforme apresentado na Eq. 7.4.

$$n = \frac{Z^2 \cdot p \cdot (1-p)}{e^2} \tag{7.1}$$

$$n_{ajustado} = \frac{n}{1 + \left(\frac{n-1}{N}\right)} \tag{7.2}$$

$$n = \frac{1{,}96^2 \cdot 0{,}5 \cdot (1-0{,}5)}{0{,}05^2} = \frac{3{,}8416 \cdot 0{,}25}{0{,}0025} = 384{,}16 \tag{7.3}$$

$$n_{ajustado} = \frac{384{,}16}{1 + \left(\frac{384{,}16-1}{154407}\right)} \approx 383{,}21 \tag{7.4}$$

7.3.8 Configuração do Experimento

Os resumos automáticos foram gerados em 35 rodadas para cada uma das 421 notícias de forma independente, resultando em 14.735 resumos automáticos por método.

Neste experimento, as métricas ROUGE-N (LIN, 2004), ROUGE-L (LIN, 2004), BLEU (PAPINENI et al., 2002), METEOR (LAVIE; AGARWAL, 2007) e BERTScorcore (ZHANG et al., 2020) utilizadas para mensuração da qualidade dos resumos automáticos foram descritas na Subseção 2.7.1. Elas foram aplicadas para mensurar a qualidade dos resumos automáticos gerados pelos métodos avaliados, utilizando como referência os resumos escritos por humanos.

Para os métodos extrativos, uma etapa de pré-processamento foi necessária, na qual as palavras foram padronizadas por meio de *stemming*, um processo que transforma as palavras em seu radical, diminuindo a variação e a complexidade linguística, e preservando o significado essencial, a raiz da palavra.

O tamanho máximo dos resumos automáticos foi limitado ao tamanho médio dos resumos de referência. Para os métodos extrativos, foi estipulado o tamanho de até 5 frases, de acordo com o tamanho médio de frases dos resumos de referência de 4.92 frases. Para os resumos abstrativos, os resumos de referência foram tokenizados e a média de tokens foi calculada, e foi fixado um resumo mínimo de $max(5, media\_tokens - valor\_tolerncia)$, e o tamanho máximo foi fixado na $media\_tokens + valor\_tolerancia$. O valor de tolerância foi definido como $(media\_tokens * 0.1)$. o processo de tokenização é a quebra de textos em unidades de sub-palavras, chamadas de tokens. O processo foi automatizado, utilizando o tokenizador do método em execução. A limitação do tamanho do resumo automático busca garantir uma comparação mais justa, conforme (NIST, 2025).

7.3.9 Instrumentação

Os seguintes materiais e recursos foram empregados no experimento:

- Google Sheets;
- Base de dados curada (2.9.1);
- Google Colab [1];
- Python (3.11.13)[2];
- Bibliotecas Python: accelerate (1.9.0), bert-score (0.3.12), bitsandbytes (0.47.0), datasets (4.0.0), evaluate (0.4.5), matplotlib (3.10.3), openpyxl (3.1.4), packaging (25.0), pandas (2.2.3), polars (1.32.0), protobuf (6.31.1), pyarrow (20.0.0), python-dotenv (1.1.1), rouge-score (0.1.2), seaborn (0.13.2), sentencepiece (0.1.99), sumy (0.11.0), tiktoken (0.9.0), tqdm (4.67.1), scipy (1.15.3), and transformers (4.54.0);
- Recursos computacionais do Núcleo de Processamento de Alto Desempenho (NPAD) da Universidade Federal do Rio Grande do Norte (UFRN).

7.4 Operacionalização do Experimento

Nesta seção, são descritos o processo de preparação do experimento, a execução, e avaliação dos resultados.

[1] https://colab.google/
[2] https://www.python.org/

7.4.1 Preparação do Experimento

A base de dados contendo todas as notícias foi obtida conforme descrito na Seção 2.9.1. Os resumos de referência, utilizados como padrão de comparação, foram elaborados por uma pesquisadora independente, de modo que, para cada notícia relacionada à saúde com indícios de irregularidade, um resumo correspondente foi produzido.

Antes da aplicação dos métodos de sumarização, foi criado um ambiente virtual para o gerenciamento de dependências, a fim de garantir reprodutibilidade e compatibilidade em diferentes ambientes de desenvolvimento. Nesse ambiente, foram instaladas todas as bibliotecas necessárias para a execução dos métodos.

No caso da geração de resumos automáticos com os métodos extrativos, foi realizada uma etapa de pré-processamento, na qual os textos das notícias foram normalizados por meio de *stemming*, reduzindo a variação e a complexidade linguística, sem comprometer o sentido essencial das palavras.

Para assegurar a execução sistemática do processo, desenvolveu-se um pipeline de geração de resumos. Esse pipeline consiste em um script capaz de receber um ou mais métodos e produzir resumos automáticos repetidamente, sendo o número de repetições (N) definido como 35 neste experimento.

Como estudo piloto, o pipeline foi inicialmente testado em 5 rodadas com 25 amostras, a fim de verificar seu funcionamento. As adaptações necessárias e eventuais falhas foram corrigidas nessa etapa preliminar. Em seguida, o processo completo foi executado para todos os métodos.

Tabela 32 – Hiperparâmetros dos modelos abstrativos.

Hiperparâmetro	**Value**
`truncation`	True
`padding`	"longest"
`return_tensors`	"pt"
`do_sample`	True
`top_k`	100
`top_p`	0.95
`temperature`	1.0
`num_return_sequences`	1

7.4.2 Execução do Experimento

A execução do experimento consistiu em duas etapas principais: geração e avaliação dos resumos automáticos.

Na etapa de geração, o pipeline foi executado para cada método utilizando a base de dados de notícias com indícios de fraude. O processo foi repetido 35 vezes por método, produzindo 35 resumos para cada uma das 421 amostras, totalizando 14.735 resumos automáticos por método avaliado.

A etapa de geração envolveu a implementação de um pipeline em Python, automatizando a execução de cada método por 35 iterações. O pipeline para os métodos extrativos é apresentado no Algoritmo 5, enquanto o pipeline para os métodos abstrativos é mostrado no Algoritmo 6. Os materiais e recursos utilizados estão detalhados na Subseção 7.3.9.

Algorithm 5 *Pipeline* de Sumarização de Texto Extrativa.

```
df[news_content_prep] ← preprocessing(df[news_content])
n_round ← 35
dfs_por_round ← [ ]
apply_rounds ← True
if apply_rounds = True then
    for round ← 1 to n_round do
        for summarizer ∈ summarizers do
            df_round ← copy(df)
            df_round[summary] ← summarizer(df_round[news_content_prep])
            df_round[method] ← nome_modelo
            df_round[round] ← rodada
            append(dfs_por_round, df_round)
        end for
        df_final ← concat(dfs_por_round)
    end for
else
    df_final ← save_results
end if=0
```

Algorithm 6 *Pipeline* de Sumarização de Texto Abstrativa.

```
n_round ← 35
for model_name ∈ models_to_use do
    model_args ← generate_args(model_name)
    save_path ← create_output_folder(model_name)
    for round ← 1 to n_rounds do
        df_with_sum ← dataset.map(summarize(model_args) )
        write_parquet(df_round, out_file)

    end for
    free_memory(model, tokenizer)
end for=0
```

Após a geração dos resumos, iniciou-se a etapa de avaliação, que consistiu na mensuração da qualidade dos resumos automáticos por meio das métricas ROUGE-N (LIN, 2004), ROUGE-L (LIN, 2004), BLEU (PAPINENI et al., 2002), METEOR (LAVIE; AGARWAL, 2007) e BERTScore (ZHANG et al., 2020), utilizando os resumos de referência como *ground truth*. Cada um dos 14.735 resumos gerados por método foi comparado com o correspondente resumo de referência elaborado manualmente. A Figura 13 ilustra o processo de execução.

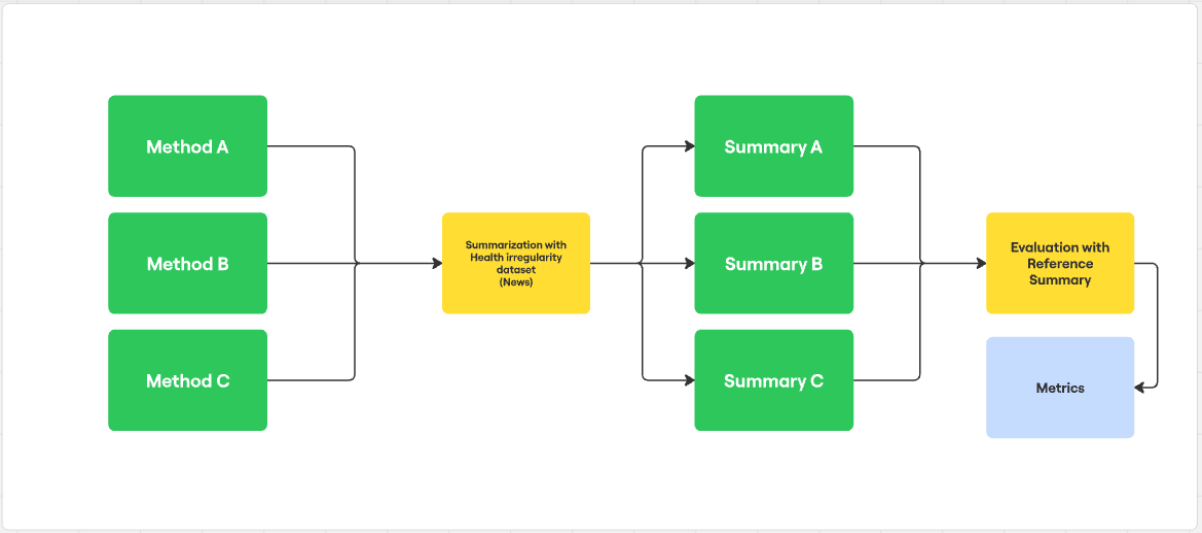


Figura 13 – Processo de Resumo e Avaliação de Textos.

7.4.3 Validação dos Dados

5 (cinco) tipos de testes estatísticos foram utilizados para análise, interpretação e validação: Anderson-Darling (AD Test), Kolmogorov-Smirnov (KS Test), Wilcoxon Signed-Rank Test (pairwise), Z-score e Intervalo Interquartil (IQR). Os testes Anderson-Darling (AD Test), Kolmogorov-Smirnov (KS Test) foram utilizados para testar a normalidade dos dados, enquanto que Wilcoxon Signed-Rank Test (pairwise) foi utilizado para comparar as medianas das métricas de ROUGE-1, ROUGE-2, ROUGE-L, BLEU, METEOR e BERTscore. As Métricas de Z-score e Intervalo Interquartil (IQR) foram utilizadas para identificação de outliers nas métricas.

7.5 Resultados

Nesta seção, são descritos o processo de Análise de Dados e Interpretação, Ameaças à Validade e Conclusões e Trabalhos Futuro.

7.5.1 Análise e Interpretação dos Dados

Para responder as perguntas listadas na Seção 7.3.4, seguiu-se a etapa de Execução e os resultados para as métricas de avaliação foram obtidos. A Tabela 33 traz os resultados das métricas ROUGE-1 (33a), ROUGE-2 (33b), ROUGE-L (33c), BLEU (33d), METEOR (33e) e BERTScore (33f) agregados por média, valor mínimo, valor máximo e desvio padrão para cada uma das métricas. A Figura 14 representa visualmente as métricas como um mapa de calor (heatmap), em que quanto mais clara a cor, mais alto o valor da métrica.

(a) Resultados para a métrica ROUGE-1 F1.

Método	Média	Mediana	Min	Max	Desvio Padrão
Qwen/Qwen2.5-7B-Instruct	0.5296	0.5310	0.2135	0.7638	0.0860
NousResearch/Hermes-3-Llama-3.2-3B	0.5270	0.5350	0.1618	0.7873	0.0930
meta-llama/Llama-3.2-3B-Instruct	0.5269	0.5309	0.2198	0.7279	0.0833
sumbasic	0.4801	0.4892	0.1597	0.7451	0.1018
Qwen/Qwen2.5-1.5B-Instruct	0.4122	0.4494	0.0000	0.7372	0.1547
TucanoBR/Tucano-2b4-Instruct	0.3796	0.3950	0.0000	0.6792	0.1079
Qwen/Qwen2.5-0.5B-Instruct	0.3590	0.3588	0.0000	0.6783	0.0992
lexrank	0.3496	0.3459	0.0333	0.7644	0.1243
TucanoBR/Tucano-1b1-Instruct	0.3464	0.3475	0.0000	0.6460	0.0961
google/gemma-2b-it	0.3296	0.3952	0.0000	0.7244	0.2063
lsa	0.3132	0.3089	0.0559	0.6636	0.1011
nicholasKluge/TeenyTinyLlama-460m	0.2894	0.2971	0.0000	0.6196	0.0951
klsum	0.2656	0.2624	0.0000	0.5796	0.1099
textrank	0.2544	0.2541	0.0245	0.5539	0.0882
maritaca-ai/sabia-7b	0.2474	0.2538	0.0000	0.6797	0.1650

(b) Resultados para a métrica ROUGE-2 F1.

Método	Média	Mediana	Min	Max	Desvio Padrão
NousResearch/Hermes-3-Llama-3.2-3B	0.2825	0.2835	0.0177	0.6411	0.0972
Qwen/Qwen2.5-7B-Instruct	0.2758	0.2709	0.0143	0.6429	0.0932
meta-llama/Llama-3.2-3B-Instruct	0.2735	0.2678	0.0287	0.5809	0.0880
sumbasic	0.2518	0.2500	0.0135	0.5743	0.1049
Qwen/Qwen2.5-1.5B-Instruct	0.1877	0.1897	0.0000	0.5745	0.1024
lexrank	0.1469	0.1278	0.0000	0.5830	0.1035
google/gemma-2b-it	0.1382	0.1319	0.0000	0.5317	0.1268
TucanoBR/Tucano-2b4-Instruct	0.1318	0.1258	0.0000	0.5000	0.0776
Qwen/Qwen2.5-0.5B-Instruct	0.1188	0.1073	0.0000	0.4842	0.0770
lsa	0.1064	0.0926	0.0000	0.4569	0.0808
TucanoBR/Tucano-1b1-Instruct	0.0999	0.0902	0.0000	0.4351	0.0662
maritaca-ai/sabia-7b	0.0851	0.0610	0.0000	0.5118	0.0909
klsum	0.0741	0.0574	0.0000	0.3704	0.0696
textrank	0.0736	0.0639	0.0000	0.3448	0.0533
nicholasKluge/TeenyTinyLlama-460m	0.0612	0.0478	0.0000	0.4136	0.0521

(c) Resultados para a métrica ROUGE-L.

Método	Média	Mediana	Min	Max	Desvio Padrão
NousResearch/Hermes-3-Llama-3.2-3B	0.3418	0.3376	0.1036	0.6990	0.0938
Qwen/Qwen2.5-7B-Instruct	0.3348	0.3246	0.1268	0.6533	0.0886
meta-llama/Llama-3.2-3B-Instruct	0.3303	0.3187	0.1423	0.6754	0.0860
sumbasic	0.3165	0.3136	0.0881	0.6209	0.1001
Qwen/Qwen2.5-1.5B-Instruct	0.2496	0.2551	0.0000	0.6423	0.1050
lexrank	0.2382	0.2261	0.0333	0.5571	0.0903
lsa	0.2099	0.1977	0.0466	0.5522	0.0708
TucanoBR/Tucano-2b4-Instruct	0.2091	0.2042	0.0000	0.5370	0.0705
google/gemma-2b-it	0.2060	0.2239	0.0000	0.5950	0.1307
Qwen/Qwen2.5-0.5B-Instruct	0.2028	0.1916	0.0000	0.5550	0.0661
TucanoBR/Tucano-1b1-Instruct	0.1803	0.1729	0.0000	0.4733	0.0549
textrank	0.1752	0.1704	0.0228	0.4412	0.0566
klsum	0.1739	0.1705	0.0000	0.4706	0.0733
nicholasKluge/TeenyTinyLlama-460m	0.1565	0.1550	0.0000	0.4167	0.0497
maritaca-ai/sabia-7b	0.1421	0.1433	0.0000	0.5564	0.0979

(d) Resultados para a métrica BLEU.

Método	Média	Mediana	Min	Max	Desvio Padrão
NousResearch/Hermes-3-Llama-3.2-3B	0.1512	0.1475	0.0053	0.4962	0.0796
meta-llama/Llama-3.2-3B-Instruct	0.1426	0.1344	0.0056	0.4693	0.0733
Qwen/Qwen2.5-7B-Instruct	0.1407	0.1351	0.0059	0.4522	0.0766
sumbasic	0.1297	0.1266	0.0015	0.3923	0.0801
Qwen/Qwen2.5-1.5B-Instruct	0.0806	0.0685	0.0000	0.4298	0.0661
google/gemma-2b-it	0.0674	0.0404	0.0000	0.4446	0.0765
lexrank	0.0674	0.0445	0.0001	0.4253	0.0666
TucanoBR/Tucano-2b4-Instruct	0.0511	0.0379	0.0000	0.3600	0.0447
Qwen/Qwen2.5-0.5B-Instruct	0.0495	0.0337	0.0000	0.3694	0.0470
lsa	0.0474	0.0309	0.0022	0.3533	0.0481
TucanoBR/Tucano-1b1-Instruct	0.0350	0.0212	0.0000	0.3087	0.0375
maritaca-ai/sabia-7b	0.0329	0.0134	0.0000	0.3511	0.0510
klsum	0.0277	0.0124	0.0000	0.3187	0.0393
textrank	0.0256	0.0147	0.0011	0.1964	0.0285
nicholasKluge/TeenyTinyLlama-460m	0.0219	0.0120	0.0000	0.2662	0.0276

(e) Resultados para a métrica METEOR.

Método	Média	Mediana	Min	Max	Desvio Padrão
NousResearch/Hermes-3-Llama-3.2-3B	0.3494	0.3448	0.0793	0.6754	0.1007
meta-llama/Llama-3.2-3B-Instruct	0.3461	0.3366	0.0985	0.6782	0.0963
Qwen/Qwen2.5-7B-Instruct	0.3400	0.3312	0.0980	0.6957	0.0972
sumbasic	0.3007	0.2908	0.0400	0.6983	0.1148
Qwen/Qwen2.5-1.5B-Instruct	0.2390	0.2420	0.0000	0.6947	0.1169
lexrank	0.2178	0.2042	0.0125	0.6407	0.1124
lsa	0.2077	0.1938	0.0520	0.5133	0.0879
TucanoBR/Tucano-2b4-Instruct	0.2051	0.2008	0.0000	0.5700	0.0843
Qwen/Qwen2.5-0.5B-Instruct	0.1946	0.1813	0.0000	0.6275	0.0834
google/gemma-2b-it	0.1932	0.2100	0.0000	0.6630	0.1486
TucanoBR/Tucano-1b1-Instruct	0.1761	0.1663	0.0000	0.5314	0.0701
textrank	0.1756	0.1637	0.0323	0.5115	0.0747
nicholasKluge/TeenyTinyLlama-460m	0.1374	0.1338	0.0000	0.5027	0.0630
klsum	0.1362	0.1247	0.0000	0.5361	0.0854
maritaca-ai/sabia-7b	0.1290	0.1095	0.0000	0.6382	0.1088

(f) Resultados para a métrica BERTScore F1.

Método	Média	Mediana	Min	Max	Desvio Padrão
Qwen/Qwen2.5-7B-Instruct	0.7115	0.7114	0.5404	0.8646	0.0525
meta-llama/Llama-3.2-3B-Instruct	0.7065	0.7089	0.5330	0.8548	0.0482
NousResearch/Hermes-3-Llama-3.2-3B	0.7058	0.7102	0.5086	0.8505	0.0511
sumbasic	0.6788	0.6848	0.4802	0.8205	0.0582
Qwen/Qwen2.5-1.5B-Instruct	0.6377	0.6470	0.4184	0.8793	0.0704
TucanoBR/Tucano-2b4-Instruct	0.6146	0.6248	0.2273	0.7964	0.0783
lexrank	0.6090	0.6074	0.3028	0.8276	0.0761
Qwen/Qwen2.5-0.5B-Instruct	0.6027	0.6019	0.3337	0.8031	0.0633
TucanoBR/Tucano-1b1-Instruct	0.5932	0.5963	0.2574	0.7688	0.0612
lsa	0.5886	0.5917	0.4159	0.7730	0.0646
nicholasKluge/TeenyTinyLlama-460m	0.5612	0.5685	0.2039	0.7445	0.0647
textrank	0.5568	0.5552	0.4025	0.7337	0.0536
google/gemma-2b-it	0.5544	0.6254	0.1784	0.8563	0.1752
klsum	0.5445	0.5522	0.1199	0.7367	0.0847
maritaca-ai/sabia-7b	0.4976	0.5178	0.2273	0.7901	0.1257

Tabela 33 – Resultados das métricas de avaliação.

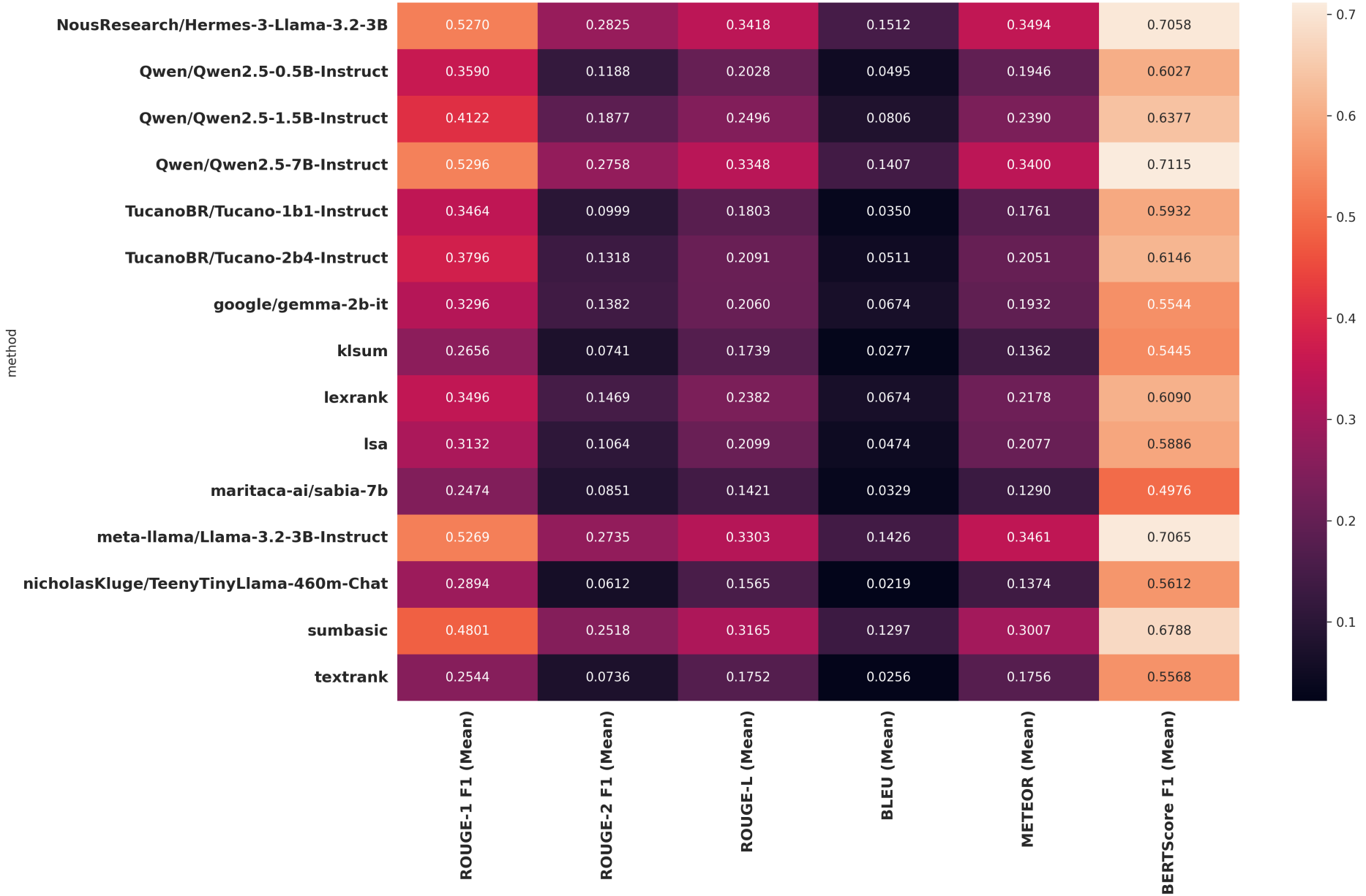


Figura 14 – Mapa de calor dos resultados das métricas de avaliação.

Diante destes resultados, nota-se que os modelos abstrativos lideraram os resultados em quase todas as métricas com um desempenho consistente, demonstrando sua capacidade de capturar as informações essenciais da base de dados de texto original se equiparando ao desempenho humano, principalmente sob a métrica BERTScore.

Apesar do desempenho inferior, os modelos extrativos ainda podem ser úteis, especialmente quando a interpretabilidade dos resultados é um fator crítico. Esses modelos empregam mecanismos estatísticos simples, como frequência de palavras no texto (SumBasic), divergência KL entre distribuições (KL-Sum), grafos de similaridade (LexRank) e tópicos latentes derivados via SVD (LSA), fornecendo, assim, regras de decisão determinísticas e verificáveis que esclarecem por que cada sentença incluída no resumo foi selecionada. Em um contexto de auditoria, onde transparência, rastreabilidade e justificativa são essenciais, tais características auxiliam os auditores na tomada de decisões informadas sobre assuntos sensíveis. Dentro da classe de modelos extrativos, o método SumBasic se destaca, classificando-se como o quarto melhor método geral de acordo com a Tabela 39, considerando seu desempenho em todas as métricas em comparação com os demais.

Entre os métodos com melhores resultados gerais, estão os modelos abstrativos Hermes-3-Llama-3.2-3B, Qwen2.5-7B-Instruct e Llama-3.2-3B-Instruct, que se destacam sob todas as métricas, a exemplo, BERTScore conforme a Tabela 39. Este resultado sugere que eles capturam e preservam melhor o significado contextual do texto original.

Para avaliar a consistência de cada resumo automático em cada amostra e em todas as iterações, analisou-se o desvio padrão das métricas, revelando a estabilidade do desempenho dos métodos em relação às métricas avaliadas. Um desvio padrão alto indica que o desempenho do modelo varia consideravelmente entre as execuções de teste, enquanto um desvio padrão baixo indica maior consistência, fornecendo resultados semelhantes independentemente do exemplo avaliado ou do número de iterações.

Para fins de comparação, como todas as métricas variam de zero a um, atribuímos as classes "Low"($std < 0.05$), "Moderate"($std > 0.05 e < 0.1$) e "High"($std > 0.1$) baseado no valor do desvio padrão para cada uma das métricas. Sob essa ótica, os métodos NousResearch/Hermes-3-Llama-3.2-3B, Qwen/Qwen2.5-7B-Instruct e meta-llama/Llama-3.2-3B-Instruct mantêm seu destaque variação moderado, conforme a Tabela 34.

Somado a isso, a fim de verificar a consistência dos resultados dos métodos, foi realizada uma análise para identificar e caracterizar a presença de outliers, considerando como tais os valores superiores a três desvios padrão, bem como aqueles identificados por meio do Intervalo Interquartil (IQR). Entre as 75 combinações de modelos e métricas, apenas 22 apresentaram outliers superiores a 1%, indicando a baixa presença de outliers, ou seja, corroborando com a consistência dos resultados entre diferentes rodadas. A Tabela 35 apresenta a distribuição dos outliers por método e métrica.

Tabela 34 – Classificação da consistência de desempenho usando o desvio padrão dos resultados.

Método	ROUGE-1 F1 (Std)	ROUGE-2 F1 (Std)	ROUGE-L (Std)	BLEU (Std)	METEOR (Std)	BERTScore F1 (Std)
NousResearch/Hermes-3-Llama-3.2-3B	Moderate	Moderate	Moderate	Moderate	High	Moderate
Qwen/Qwen2.5-0.5B-Instruct	Moderate	Moderate	Moderate	Low	Moderate	Moderate
Qwen/Qwen2.5-1.5B-Instruct	High	High	High	Moderate	High	Moderate
Qwen/Qwen2.5-7B-Instruct	Moderate	Moderate	Moderate	Moderate	Moderate	Moderate
TucanoBR/Tucano-1b1-Instruct	Moderate	Moderate	Moderate	Low	Moderate	Moderate
TucanoBR/Tucano-2b4-Instruct	High	Moderate	Moderate	Low	Moderate	Moderate
google/gemma-2b-it	High	High	High	Moderate	High	High
klsum	High	Moderate	Moderate	Low	Moderate	Moderate
lexrank	High	High	Moderate	Moderate	High	Moderate
lsa	High	Moderate	Moderate	Low	Moderate	Moderate
maritaca-ai/sabia-7b	High	Moderate	Moderate	Moderate	High	High
meta-llama/Llama-3.2-3B-Instruct	Moderate	Moderate	Moderate	Moderate	Moderate	Low
nicholasKluge/TeenyTinyLlama-460m-Chat	Moderate	Moderate	Low	Low	Moderate	Moderate
sumbasic	High	High	High	Moderate	High	Moderate
textrank	Moderate	Moderate	Moderate	Low	Moderate	Moderate

Tabela 35 – Percentagem de outliers detectados pelo escore Z e pelo intervalo interquartil (IQR) para diferentes métodos e métricas. Filtrados apenas os valores acima de 1%.

Método	Métrica	N	Z_Outliers_%	IQR_Outliers_%
nicholasKluge/TeenyTinyLlama-460m-Chat	BLEU	14735	2.586	10.499
TucanoBR/Tucano-1b1-Instruct	BLEU	14735	2.260	7.112
klsum	ROUGE-2 F1	14735	2.138	2.850
klsum	BLEU	14735	2.138	6.888
maritaca-ai/sabia-7b	BLEU	14735	2.029	12.257
textrank	BLEU	14735	1.900	8.789
maritaca-ai/sabia-7b	ROUGE-2 F1	14735	1.887	4.269
nicholasKluge/TeenyTinyLlama-460m-Chat	ROUGE-2 F1	14735	1.805	4.466
textrank	METEOR	14735	1.663	2.613
klsum	BERTScore F1	14735	1.663	5.701
textrank	ROUGE-2 F1	14735	1.663	1.900
TucanoBR/Tucano-2b4-Instruct	ROUGE-1 F1	14735	1.636	1.934
Qwen/Qwen2.5-0.5B-Instruct	BLEU	14735	1.513	3.509
TucanoBR/Tucano-2b4-Instruct	BERTScore F1	14735	1.513	3.101
klsum	METEOR	14735	1.425	3.325
lsa	ROUGE-2 F1	14735	1.425	1.425
lexrank	METEOR	14735	1.425	3.800
lexrank	BLEU	14735	1.425	1.663
maritaca-ai/sabia-7b	METEOR	14735	1.391	1.778
TucanoBR/Tucano-2b4-Instruct	BLEU	14735	1.344	2.246
Qwen/Qwen2.5-1.5B-Instruct	BLEU	14735	1.289	2.328
google/gemma-2b-it	BLEU	14735	1.140	1.608
nicholasKluge/TeenyTinyLlama-460m-Chat	METEOR	14735	1.086	5.375

Para analisar a redução da sobrecarga de informação, além das métricas de qualidade, foram avaliadas a redução textual em relação ao documento original e seu tamanho em comparação ao desempenho humano. Além de preservar a qualidade, os melhores modelos — NousResearch/Hermes-3-Llama-3.2-3B, Qwen/Qwen2.5-7B-Instruct e meta-llama/Llama-3.2-3B-Instruct — produziram resumos com comprimentos relativamente próximos à média humana, com diferenças médias na contagem de palavras de 10,57%, 3,25% e 8,13%, respectivamente, em comparação ao desempenho humano. A Tabela 36 descreve a redução da sobrecarga de informação, apresentando o comprimento médio de todos os artigos de notícias em palavras (Orig. Words), o comprimento médio dos resumos humanos em todos os documentos (Ref. Words) e seu tamanho relativo em comparação com o comprimento médio do documento (Ref.

%), o comprimento médio dos resumos automáticos (Auto Words) e seu tamanho relativo em comparação com o comprimento médio do documento (Auto %), e a diferença relativa entre os resumos automáticos e o desempenho humano (Dif %). O Apêndice A compara amostras de resumos de referência com resumos automáticos gerados pelos modelos de melhor desempenho.

Tabela 36 – Percentage difference in information overload reduction between human summary vs. automatic summary. Ordenado por dif (%)

Method	Type	Orig. Words	Ref. Words	Auto Words	Ref. (%)	Auto (%)	Dif (%)
maritaca-ai/sabia-7b	abstractive	2726	123	90	4.51	3.30	-26.83
google/gemma-2b-it	abstractive	2726	123	102	4.51	3.74	-17.07
klsum	extractive	2726	123	113	4.51	4.15	-8.13
nicholasKluge/TeenyTinyLlama-460m-Chat	abstractive	2726	123	122	4.51	4.48	-0.81
sumbasic	extractive	2726	123	124	4.51	4.55	0.81
Qwen/Qwen2.5-7B-Instruct	abstractive	2726	123	127	4.51	4.66	3.25
TucanoBR/Tucano-1b1-Instruct	abstractive	2726	123	128	4.51	4.70	4.07
Qwen/Qwen2.5-1.5B-Instruct	abstractive	2726	123	129	4.51	4.73	4.88
TucanoBR/Tucano-2b4-Instruct	abstractive	2726	123	129	4.51	4.73	4.88
Qwen/Qwen2.5-0.5B-Instruct	abstractive	2726	123	131	4.51	4.81	6.50
meta-llama/Llama-3.2-3B-Instruct	abstractive	2726	123	133	4.51	4.88	8.13
NousResearch/Hermes-3-Llama-3.2-3B	abstractive	2726	123	136	4.51	4.99	10.57
lexrank	extractive	2726	123	171	4.51	6.27	39.02
lsa	extractive	2726	123	212	4.51	7.78	72.36
textrank	extractive	2726	123	290	4.51	10.64	135.77

Apesar dos bons resultados, não é possível fazer afirmações sem evidências estatísticas suficientemente conclusivas, e, para podermos-fazer afirmações comparativas, precisamos ter evidências estatísticas suficientemente conclusivas. Portanto, um nível de significância (α) de 0.05 foi definido para todo o experimento. Foram aplicados testes de normalidade

Um teste de normalidade foi aplicado utilizando métodos robustos para grandes amostras (14.735 por método), especificamente Anderson-Darling (AD Test) e Kolmogorov-Smirnov (KS Test), com o objetivo de identificar qual teste de hipótese seria mais adequado para responder às Perguntas de Pesquisa (7.3.4). O AD Test foi utilizado como métrica principal, enquanto o KS Test serviu como métrica complementar de verificação. Os resultados indicaram que não há evidências de que qualquer um dos conjuntos de dados siga uma distribuição normal, conforme apresentado nas Tabelas 37 e 38. Para as 14735 amostras, o valor crítico para a Estaística AD foi de 0.787, ou seja, rejeitando para todos os métodos em todas as métricas. Ao analisar o valor-p para o teste Kolmogorov-Smirnov, confirmou-se que nenhuma distribuição de resultados para nenhuma das métricas segue uma distribuição aproximadamente normal. Dessa forma, tornou-se necessário o emprego de testes de hipótese não paramétricos. Como principal abordagem, utilizou-se o teste Wilcoxon Signed-Rank Test, aplicado de forma pareada (Método A vs. Método B) com o objetivo de avaliar se a distribuição dos valores do "Método A"é significativamente superior à do "Método B".

Tabela 37 – Resultados do Teste de Normalidade — Anderson-Darling (AD_Statistic)

Método	**BERTScore F1**	**BLEU**	**METEOR**	**ROUGE-1 F1**	**ROUGE-2 F1**	**ROUGE-L**
NousResearch/Hermes-3-Llama-3.2-3B	29.77	56.15	42.39	69.31	5.49	29.40
Qwen/Qwen2.5-0.5B-Instruct	3.92	674.17	138.18	6.38	153.81	156.32
Qwen/Qwen2.5-1.5B-Instruct	126.35	296.25	56.14	594.46	42.02	125.39
Qwen/Qwen2.5-7B-Instruct	3.70	89.42	39.92	10.83	19.21	68.15
TucanoBR/Tucano-1b1-Instruct	40.96	1116.53	118.44	9.26	168.26	159.44
TucanoBR/Tucano-2b4-Instruct	235.25	491.28	26.11	112.81	56.36	55.47
google/gemma-2b-it	657.38	747.67	238.79	486.87	451.69	205.86
klsum	298.46	1407.12	199.69	47.79	451.36	183.15
lexrank	12.16	674.93	164.17	9.18	200.53	136.52
lsa	33.69	670.52	139.81	11.45	230.39	110.26
maritaca-ai/sabia-7b	112.57	1666.28	244.46	132.77	613.58	126.77
meta-llama/Llama-3.2-3B-Instruct	9.32	76.45	55.90	23.79	37.85	105.51
nicholasKluge/TeenyTinyLlama-460m-Chat	107.45	1602.94	119.82	83.45	452.51	157.85
sumbasic	43.29	83.20	27.09	42.31	10.65	25.12
textrank	8.70	1179.22	145.10	17.34	236.46	62.60

Tabela 38 – Resultados do Teste de Normalidade — Kolmogorov-Smirnov (KS_pvalue)

Method	**BERTScore F1**	**BLEU**	**METEOR**	**ROUGE-1 F1**	**ROUGE-2 F1**	**ROUGE-L**
NousResearch/Hermes-3-Llama-3.2-3B	2.98×10^{-17}	2.19×10^{-40}	4.11×10^{-19}	2.37×10^{-38}	1.66×10^{-6}	1.44×10^{-23}
Qwen/Qwen2.5-0.5B-Instruct	2.95×10^{-2}	3.76×10^{-307}	5.94×10^{-56}	5.03×10^{-4}	5.01×10^{-53}	1.76×10^{-73}
Qwen/Qwen2.5-1.5B-Instruct	5.80×10^{-54}	1.75×10^{-159}	5.92×10^{-29}	9.12×10^{-240}	1.68×10^{-22}	9.67×10^{-61}
Qwen/Qwen2.5-7B-Instruct	7.49×10^{-2}	1.14×10^{-31}	5.13×10^{-19}	1.99×10^{-10}	5.00×10^{-10}	1.20×10^{-33}
TucanoBR/Tucano-1b1-Instruct	3.44×10^{-28}	0	1.24×10^{-61}	2.34×10^{-7}	3.54×10^{-63}	5.36×10^{-81}
TucanoBR/Tucano-2b4-Instruct	3.59×10^{-92}	2.09×10^{-205}	1.33×10^{-16}	2.34×10^{-60}	1.06×10^{-26}	3.25×10^{-23}
google/gemma-2b-it	0	0	1.93×10^{-120}	2.19×10^{-206}	1.60×10^{-260}	1.20×10^{-127}
klsum	6.99×10^{-126}	0	2.19×10^{-121}	6.50×10^{-29}	1.63×10^{-265}	7.69×10^{-82}
lexrank	6.23×10^{-16}	4.71×10^{-314}	1.43×10^{-78}	1.40×10^{-10}	4.59×10^{-95}	9.75×10^{-126}
lsa	1.04×10^{-23}	0	9.79×10^{-69}	4.76×10^{-8}	8.79×10^{-114}	4.51×10^{-70}
maritaca-ai/sabia-7b	9.04×10^{-76}	0	7.66×10^{-179}	1.02×10^{-57}	0	2.24×10^{-81}
meta-llama/Llama-3.2-3B-Instruct	1.47×10^{-7}	5.38×10^{-40}	4.28×10^{-32}	4.33×10^{-9}	1.26×10^{-20}	6.89×10^{-57}
nicholasKluge/TeenyTinyLlama-460m-Chat	1.09×10^{-55}	0	1.14×10^{-48}	2.22×10^{-35}	1.07×10^{-185}	8.33×10^{-58}
sumbasic	6.03×10^{-42}	9.10×10^{-40}	3.53×10^{-22}	1.61×10^{-28}	2.03×10^{-13}	3.61×10^{-23}
textrank	5.96×10^{-7}	0	9.19×10^{-109}	1.40×10^{-12}	7.29×10^{-107}	1.10×10^{-36}

Para avaliar se um método tem o resultado significativamente maior que o outro, foi utilizado o teste Wilcoxon Signed-Rank Test. Ele é um teste não-paramétrico para amostras pareadas, e avalia se a mediana das diferenças entre dois métodos (ou condições) é significativamente diferente de zero. Após sua aplicação, obtiveram-se evidências acerca do desempenho comparativo em cada uma das métricas de avaliação. Estes resultados foram resumidos na Tabela 39, em que se descreve a quantidade de pontuações (quando o valor-p é significativo) do método base ("Modelo A") sobre o método alternativo ("Modelo B") na comparação par a par para cada métrica, enquanto a coluna Score é o somatório das pontuações do modelo ("Modelo A") sobre os modelos alternativos ("Modelo B").

Após a identificação dos modelos com melhor desempenho comparativo, buscou-se mensurar a magnitude da diferença entre as medianas dos melhores modelos nos cenários mais favorável e menos favorável, de modo a avaliar o quanto um modelo se destaca em relação ao alternativo. A Tabela 40 apresenta o modelo base, os modelos correspondentes aos cenários de melhor e pior desempenho, bem como as respectivas métricas. Ademais, a Tabela 40 descreve os resultados obtidos quando o modelo base se mostrou superior ou inferior em cada métrica analisada.

Tabela 39 – Summary of the Wilcoxon Signed-Rank Test (pairwise). Sorted by Score

Method	ROUGE-1 F1	ROUGE-2 F1	ROUGE-L	BLEU	METEOR	BERTScore F1	Score
NousResearch/Hermes-3-Llama-3.2-3B	12	14	14	14	14	12	80
Qwen/Qwen2.5-7B-Instruct	13	13	13	12	12	14	77
meta-llama/Llama-3.2-3B-Instruct	12	12	12	13	13	12	74
sumbasic	11	11	11	11	11	11	66
Qwen/Qwen2.5-1.5B-Instruct	10	10	10	10	10	10	60
lexrank	6	9	9	9	9	8	50
TucanoBR/Tucano-2b4-Instruct	9	7	6	7	7	9	45
Qwen/Qwen2.5-0.5B-Instruct	8	6	5	6	6	7	38
google/gemma-2b-it	5	8	6	8	5	4	36
lsa	4	5	6	5	7	5	32
TucanoBR/Tucano-1b1-Instruct	6	4	4	4	3	6	27
textrank	1	2	3	2	3	2	13
nicholasKluge/TeenyTinyLlama-460m-Chat	3	0	1	0	2	3	9
klsum	2	1	2	1	1	1	8
maritaca-ai/sabia-7b	0	3	0	2	0	0	5

Tabela 40 – Magnitude da diferença entre as medianas dos melhores modelos no melhor e pior cenário.

Método Base	Métrica	Método Comparado	Maior dif.	Método Comparado	Menor dif.
NousResearch/Hermes-3-Llama-3.2-3B	BERTScore F1	maritaca-ai/sabia-7b	0.192399	sumbasic	0.025466
meta-llama/Llama-3.2-3B-Instruct	BERTScore F1	maritaca-ai/sabia-7b	0.191081	sumbasic	0.024147
Qwen/Qwen2.5-7B-Instruct	BERTScore F1	maritaca-ai/sabia-7b	0.193555	NousResearch/Hermes-3-Llama-3.2-3B	0.001156
NousResearch/Hermes-3-Llama-3.2-3B	BLEU	nicholasKluge/TeenyTinyLlama-460m-Chat	0.135554	Qwen/Qwen2.5-7B-Instruct	0.012369
meta-llama/Llama-3.2-3B-Instruct	BLEU	nicholasKluge/TeenyTinyLlama-460m-Chat	0.122471	Qwen/Qwen2.5-7B-Instruct	-0.000714
Qwen/Qwen2.5-7B-Instruct	BLEU	nicholasKluge/TeenyTinyLlama-460m-Chat	0.123185	sumbasic	0.008521
NousResearch/Hermes-3-Llama-3.2-3B	METEOR	maritaca-ai/sabia-7b	0.235310	meta-llama/Llama-3.2-3B-Instruct	0.008221
meta-llama/Llama-3.2-3B-Instruct	METEOR	maritaca-ai/sabia-7b	0.227089	Qwen/Qwen2.5-7B-Instruct	0.005301
Qwen/Qwen2.5-7B-Instruct	METEOR	maritaca-ai/sabia-7b	0.221789	sumbasic	0.040476
NousResearch/Hermes-3-Llama-3.2-3B	ROUGE-1 F1	maritaca-ai/sabia-7b	0.281186	sumbasic	0.045868
meta-llama/Llama-3.2-3B-Instruct	ROUGE-1 F1	maritaca-ai/sabia-7b	0.277018	sumbasic	0.041700
Qwen/Qwen2.5-7B-Instruct	ROUGE-1 F1	maritaca-ai/sabia-7b	0.277188	meta-llama/Llama-3.2-3B-Instruct	0.000170
NousResearch/Hermes-3-Llama-3.2-3B	ROUGE-2 F1	nicholasKluge/TeenyTinyLlama-460m-Chat	0.235764	Qwen/Qwen2.5-7B-Instruct	0.012609
meta-llama/Llama-3.2-3B-Instruct	ROUGE-2 F1	nicholasKluge/TeenyTinyLlama-460m-Chat	0.220045	sumbasic	0.017806
Qwen/Qwen2.5-7B-Instruct	ROUGE-2 F1	nicholasKluge/TeenyTinyLlama-460m-Chat	0.223155	meta-llama/Llama-3.2-3B-Instruct	0.003110
NousResearch/Hermes-3-Llama-3.2-3B	ROUGE-L	maritaca-ai/sabia-7b	0.194257	Qwen/Qwen2.5-7B-Instruct	0.013018
meta-llama/Llama-3.2-3B-Instruct	ROUGE-L	maritaca-ai/sabia-7b	0.175403	sumbasic	0.005116
Qwen/Qwen2.5-7B-Instruct	ROUGE-L	maritaca-ai/sabia-7b	0.181239	meta-llama/Llama-3.2-3B-Instruct	0.005836

Os modelos *NousResearch/Hermes-3-Llama-3.2-3B* e *Qwen/Qwen2.5-7B-Instruct* foram pré-treinados utilizando bases de dados no idioma inglês, enquanto o *meta-llama/Llama-3.2-3B-Instruct* foi treinado em bases multilíngues (Tabela 30). Apesar do treinamento totalmente ou majoritariamente em língua estrangeira, esses modelos apresentaram o melhor desempenho em todos os critérios de avaliação de qualidade de resumo e nas comparações par a par. Para a

métrica BERTScore, a diferença foi de 0.19 ou 19%, ou seja, comparado ao desempenho humano, os três melhores modelos são 19% melhores que o modelo comparado. Em contrapartida, os modelos com piores resultados relativos aos três melhores foram os pré-treinados em português, *maritaca-ai/sabia-7b* e *nicholasKluge/TeenyTinyLlama-460m-Chat*. Já os modelos que exibiram menor diferença de desempenho foram o extrativo *sumbasic* e os abstrativos *NousResearch/Hermes-3-Llama-3.2-3B* e *Qwen/Qwen2.5-7B-Instruct*.

Os resultados de todas as avaliações indicam, sob a perspectiva de um auditor, que esses modelos são tecnicamente confiáveis, pois apresentam consistência nos resultados, destacam as informações mais relevantes e mantêm um tamanho médio de resumo comparável ao desempenho humano. Dessa forma, por meio de métodos automatizados, ao contribuir para a redução da sobrecarga informacional, esses modelos podem dar suporte ao processo de auditoria na fase analítica, aumentando a eficácia e efetividade no levantamento de informações, preparando as equipes para a fase operativa em menor tempo.

7.6 Considerações Finais

O processo de auditoria caracteriza-se, em geral, por ser custoso, demorado e demandar substanciais recursos humanos e materiais. Nesse sentido, torna-se necessário implementar soluções e técnicas que possibilitem a automatização da análise de denúncias de corrupção. Esse processo é usualmente dividido em duas etapas: na primeira, busca-se identificar elementos e evidências de corrupção, tais como fornecedores, contratos, funcionários, clientes e demais partes interessadas, avaliando a plausibilidade e consistência das denúncias e indícios de fraude; na segunda etapa, desenvolve-se a investigação propriamente dita.

Para a construção do conhecimento necessário à atividade de auditoria, é imprescindível o levantamento de informações relacionadas ao objetivo da auditoria. Nessa fase, recorre-se a diversas fontes, incluindo páginas da internet. Para apoiar o processo de coleta de informações, podem ser aplicadas técnicas de *webscraping* voltadas à extração massiva de dados em sites do contexto da saúde. Ademais, para auxiliar na análise dessa grande quantidade de dados, técnicas de PLN, como a sumarização de texto, podem ser empregadas, reduzindo significativamente o tempo e os recursos necessários para a análise e a coleta de evidências de possíveis irregularidades
.

Neste contexto, com o objetivo de apoiar, aprimorar e otimizar a coleta de informações relevantes que possam auxiliar no combate a irregularidades, este trabalho apresenta os resultados da aplicação de 15 métodos de sumarização automática de textos em um conjunto de notícias relacionadas à área da saúde com indícios de irregularidade. Buscou-se, assim, avaliar se tais métodos podem contribuir para o processo de auditoria, reduzindo a sobrecarga informacional, bem como identificar quais se mostram mais eficazes para essa tarefa.

Neste experimento controlado **in vitro**, utilizando uma base de dados curada com 421

amostras de notícias, foram gerados resumos automáticos por meio de 15 métodos, repetidos em 35 rodadas. Os resultados foram avaliados de forma robusta com base em múltiplas métricas de desempenho (ROUGE-N (LIN, 2004), ROUGE-L (LIN, 2004), BLEU (PAPINENI et al., 2002), METEOR (LAVIE; AGARWAL, 2007) e BERTScore (ZHANG et al., 2020)). Com os resultados sobre a qualidade do resumo sob várias métricas, analisou-se a consistência entre os resultados por meio do desvio padrão, a presença de outliers, o ganho de redução de sobrecarga informacional. Além disso, todos os métodos por meio do teste de hipótese Wilcoxon Signed-Rank Test (pareado), cujos resultados são apresentados neste estudo.

Entre os métodos com melhor desempenho relacionado à qualidade dos resumos e relativos aos demais, destacam-se os modelos **NousResearch/Hermes-3-Llama-3.2-3B**, **Qwen/Qwen2.5-7B-Instruct** e **meta-llama/Llama-3.2-3B-Instruct**. Esses métodos se sobressaem de forma consistente nas diferentes métricas de avaliação, demonstrando capacidade superior em capturar e preservar o significado contextual do texto original, além de sintetizar adequadamente as principais informações, quando comparados ao desempenho humano.

Sob a perspectiva de um auditor, esses modelos demonstram ser tecnicamente confiáveis, uma vez que apresentam consistência nos resultados, destacam as informações mais relevantes e mantêm um tamanho médio de resumo comparável ao desempenho humano. Assim, por meio de métodos automatizados e ao contribuir para a redução da sobrecarga informacional, esses modelos podem apoiar o processo de auditoria na fase analítica, aumentando a eficácia e a efetividade no levantamento de informações e possibilitando a preparação das equipes para a fase operativa em menor tempo.

Para a avaliação do experimento, torna-se necessário considerar os fatores que podem influenciar os resultados, caracterizados como ameaças à validade interna e externa.

- **Validade Interna**: O processo de classificação das notícias foi conduzido por dois anotadores. Por se tratar de uma atividade manual e intensiva, existe a possibilidade de ocorrência de falhas de classificação. Para mitigar esse risco, um terceiro avaliador interveio nos casos de discordância entre os anotadores quanto à categorização da notícia.

- **Validade Externa**: Considerando a natureza dos dados, notícias, e o fato de que apenas os títulos (*headlines*) foram utilizados neste experimento, a baixa variabilidade linguística poderia dificultar a tarefa de classificação. Para mitigar esse problema, selecionou-se os objetos do experimento de forma robusta e representativa. Além disso, foram aplicados métodos robustos de treinamento, como a validação cruzada estratificada e a utilização de uma base de teste independente em cada rodada.

Por fim, visando potencializar a redução da sobrecarga informacional, torna-se possível identificar automaticamente notícias que apresentem indícios de irregularidades e sintetizar seu conteúdo por meio do agrupamento temático, em vez de tratá-las individualmente. Com esse

propósito, os Capítulos 8 e 9 avaliam, por meio de experimentos controlados, dois métodos auxiliares: Classificação de Texto e Modelagem de Tópicos, respectivamente.

8

Avaliação Experimental de Métodos de Classificação de Texto

Este capítulo apresenta, por meio de uma reprodução parcial, o experimento controlado de métodos de classificação de texto introduzido no artigo *Experimental Evaluation of Machine Learning Algorithms for Classifying Health-Related News with Indications of Irregularity* e submetido e **aceito** na conferência *XXII Simpósio Brasileiro de Sistemas de Informação (SBSI)*.

8.1 Contextualização

A crescente produção de dados não estruturados, particularmente dados textuais, tem impulsionado a aplicação de técnicas de Processamento de Linguagem Natural (NLP) na administração pública. A análise de múltiplas fontes de informação possibilita a identificação de padrões e o desenvolvimento de modelos preditivos para otimizar estratégias, aprimorar a prestação de serviços e assegurar o monitoramento e a segurança da população.

O processo de auditoria caracteriza-se por elevados custos, longa duração e forte dependência de recursos humanos e materiais, o que demanda soluções capazes de automatizar a análise de relatórios de corrupção.

Com foco na identificação preliminar de potenciais irregularidades, propõe-se o uso de modelos de *machine learning* para apoiar o processo de auditoria, por meio da identificação de notícias relacionadas à área da saúde que possam indicar irregularidades.

Este estudo fundamenta-se na Teoria da Carga Cognitiva, ao investigar métodos para a redução da sobrecarga de informação e conduzimos um experimento controlado *in vitro* para avaliar cientificamente 54 algoritmos de *machine learning* e comparar métricas como Accuracy, Precision, Recall e F1-score. Adicionalmente, foi realizada uma análise da complexidade assintótica dos algoritmos.

O modelo *Random Forest* destacou-se em termos de efetividade, alcançando uma accuracy

de 99,90%, um recall de 98,62% e um F1-score de 99,28%, enquanto os modelos Naive Bayes e Logistic Regression sobressaíram em eficiência, apresentando complexidade linear $O(nd)$ tanto para treinamento quanto para predição, além de baixo consumo de memória.

Os resultados demonstram a viabilidade do uso de modelos de *machine learning* para a identificação de notícias relacionadas à área da saúde com potenciais irregularidades. Essa abordagem aprimora a etapa de coleta de informações e evidências de corrupção realizada por auditores da AudSUS na detecção de possíveis irregularidades, contribuindo, assim, para a eficiência da gestão de recursos públicos.

8.2 Materiais e Métodos

Este é um estudo experimental, seguindo os passos apresentados por (Colaço JÚNIOR et al., 2022; Colaço JÚNIOR, 2025) e descritos no Capítulo 1.6 para avaliação dos resultados de modelos de machine learning aplicados à classificação de texto, utilizando notícias relacionadas à saúde com indícios de irregularidade, avaliando a qualidade dos resumos utilizando as métricas Acurácia, Precision, Recall e F1.

A descrição da base de dados utilizada, o processo de seleção de notícias com indícios de irregularidade e métricas de avaliação dos resultados foi descrito no Capítulo 2.9.1.

8.3 Configuração Experimental

Nesta seção, são apresentados o objetivo da avaliação experimental, o planejamento, as perguntas de pesquisa, as variáveis independentes, as variáveis dependentes e as hipóteses.

8.3.1 Objetivo

Para formalizar o objetivo deste estudo, o modelo de *Goal Question Metric* (GQM) de (BASILI; WEISS, 1984) foi utilizado. Este trabalho tem como objetivo **analisar** modelos de machine learning por meio de um experimento controlado (in vitro), **com a finalidade de** avaliá-los, contra os resultados da base classificada por humanos, **com respeito às** métricas de accuracy, precision, recall e F1-score, **em relação à** classificação de notícias relacionadas à saúde com indício de irregularidade, **sob o ponto de vista** de Cientistas de Dados e Auditores do SUS, **no contexto de** auditorias na saúde pública do Brasil, realizadas pelo Departamento Nacional de Auditoria do SUS (AudSUS).

8.3.2 Planejamento

O experimento foi realizado em ambiente controlado *in vitro*, utilizando a base de dados descrita na Subseção 2.9.1 e modelos de machine learning para classificar as notícias. A base

utilizada filtrada contém 6.239 notícias.

8.3.3 Seleção de Contexto

Apesar dos avanços tecnológicos, diversos processos no setor público ainda dependem de buscas manuais para a construção de conhecimento. Esse cenário também se verifica no Departamento Nacional de Auditoria do Sistema Único de Saúde (AudSUS), responsável pelo controle e fiscalização do SUS. As atividades de auditoria conduzidas pelo órgão desempenham papel essencial na gestão e utilização adequada dos recursos públicos; contudo, o processo é altamente intensivo em recursos humanos devido à elevada demanda, uma vez que os auditores devem fiscalizar todas as áreas do SUS e, adicionalmente, atender às demandas internas e externas do Ministério da Saúde (FONTES et al., 2023). Nesse contexto, o experimento proposto busca apoiar a fase analítica do processo de auditoria, etapa responsável pelo planejamento e pela preparação da equipe para a fase operativa, por meio do levantamento de informações relacionadas ao objetivo da auditoria.

8.3.4 Questões de Pesquisa

Para guiar o experimento e com o objetivo de cumprir o objetivo do trabalho, foram elaboradas as seguintes questões de pesquisa:

- RQ1: Quais dos algoritmos selecionados são os melhores em termos de eficácia ?
- RQ2: Dentre os algoritmos selecionados, qual são os melhores em termos de eficiência ?

Para responder às questões de pesquisa, as seguintes hipóteses teóricas descritas na Tabela 41 foram criadas.

Tabela 41 – Questões de pesquisa e hipóteses associadas

RQ	**Hipótese nula (H_0)**	**Hipótese alternativa (H_1)**
RQ1	Os algoritmos possuem a mesma efetividade.	Os algoritmos não possuem a mesma efetividade.
RQ2	Os modelos possuem a mesma eficiência.	Os modelos não possuem eficiência.

8.3.5 Variáveis Dependentes

As variáveis dependentes, ou variáveis de saída, foram as notícias classificadas, das quais podem ser derivadas as métricas Accuracy, Precision, Recall e F1-score.

8.3.6 Variáveis Independentes

Neste experimento, as variáveis independentes ou variável de entrada, são: a base de dados de notícias com indício de irregularidade anotada (classificada); os algoritmos utilizados

para a tarefa de classificação: *AdaBoost, Catboost (CatBoostClassifier), GradientBoosting, KNN, LightGBM (LGBMClassifier), Logistic Regression, Naive Bayes (Multinomial), RandomForest, Support Vector Classification (SVC), XGBoost (XGBClassifier)*

8.3.7 Seleção de Objetos

Seguindo o contexto descrito na Subseção 8.3.3, os objetos deste experimento são as notícias relacionadas à saúde com indícios de irregularidade (Capítulo 2.9.1). Para o cálculo amostral, foi considerada uma população finita de 154.407 notícias, ou seja, o total de notícias da base de dados completa. Vale ressaltar que a amostra final excede o número estimado conforme a Eq. 8.2. Para o cálculo amostral, foi considerado um nível de confiança de 95% (valor de $Z = 1{,}96$), um erro amostral tolerável de 5% ($e = 0{,}05$) e uma proporção esperada de 50% ($p = 0{,}5$), o que maximiza a variabilidade da amostra e garante um tamanho mais conservador.

O cálculo do tamanho amostral para uma população finita foi realizado em duas etapas: primeiro, estimou-se a amostra para uma população infinita (n), utilizando a Eq.8.1, e, em seguida, aplicou-se o ajuste para população finita ($n_{ajustado}$), conforme a Eq.8.2, resultando em aproximadamente 383,21 amostras, conforme a Eq. 8.4. Por fim, foram utilizadas todas as amostras classificadas manualmente de 6.239, contendo 421 notícias da classe "Saúde irregularidade"e as demais com a classe "Saúde Genérica"ou "Saúde Genérica", que excedem a quantidade mínima de 384 notícias para uma amostra representativa.

$$n = \frac{Z^2 \cdot p \cdot (1-p)}{e^2} \tag{8.1}$$

$$n_{ajustado} = \frac{n}{1 + \left(\frac{n-1}{N}\right)} \tag{8.2}$$

$$n = \frac{1{,}96^2 \cdot 0{,}5 \cdot (1-0{,}5)}{0{,}05^2} = \frac{3{,}8416 \cdot 0{,}25}{0{,}0025} = 384{,}16 \tag{8.3}$$

$$n_{ajustado} = \frac{384{,}16}{1 + \left(\frac{384{,}16-1}{154407}\right)} \approx 383{,}21 \tag{8.4}$$

8.3.8 Configuração de Experimento

A classificação foi realizada em 35 rodadas com os 9 algoritmos selecionados. Para cada rodada, foi utilizada uma combinação de métodos de otimização de hiperparâmetros (*Default*, *random* ou bayes) com métodos de treinamento (hold out ou cross validation), gerando, assim, 6 combinações de classificadores (’*Default*’ e ’hold out’, ’*Default*’ e ’cross validation’, ’*random*’ e ’hold out’, ’*random*’ e ’cross validation’, ’bayes’e ’hold out’ ou ’bayes’ ’cross validation’) totalizando 54 modelos, executadas 35 vezes para cada um dos modelos selecionados.

Antes da execução do *pipeline* de classificação, uma etapa de preprocessamento foi necessária em que foi aplicado tf-idf, conforme (SALTON; BUCKLEY, 1988), para contabilizar a importância relativa das palavras dos títulos das notícias e criar os atributos (features) ou variáveis independentes da base de dados. Após a preparação dos dados, o *pipeline* de execução foi executado.

8.3.9 Instrumentação

Os seguintes materiais e recursos foram utilizados:

- Google Sheets;
- Base de dados anotada e com resumos de referência (2.9.1);
- Google Colab [1];
- A linguagem de programação Python (3.11.13)[2];
- Bibliotecas Python: catboost (1.2.8), lightgbm (4.6.0), matplotlib (3.10.5), numpy (2.2.6), openpyxl (3.1.4), pandas (2.3.1), pyarrow (21.0.0), scikit-learn (1.7.1), scikit-optimize (0.10.2), seaborn (0.13.2), sentence-transformers (5.1.0), tqdm (4.67.1), xgboost (3.0.4) e uv (0.8.14);
- Recursos computacionais do Núcleo de Processamento de Alto Desempenho (NPAD) da Universidade Federal do Rio Grande do Norte (UFRN).

8.4 Operacionalização do Experimento

Nesta seção, são descritos o processo de preparação do experimento, a execução e a avaliação dos resultados.

8.4.1 Preparação do Experimento

O ambiente de execução no supercomputador do NPAD foi preparado por meio da biblioteca uv, utilizada para criação do ambiente virtual e instalação das bibliotecas necessárias, descritas na Subseção 8.3.9, para a tarefa de classificação; em seguida, a base foi enviada para o ambiente.

Para assegurar a execução sistemática do processo, desenvolveu-se um *pipeline* de classificação. Esse *pipeline* consiste em um script que executa as etapas para cada modelo candidato descrito na Subseção 8.3.6 e para cada uma das 6 combinações de tipos de classificadores

[1] https://colab.google/
[2] https://www.python.org/

descritas na Subseção 8.3.8. O Algoritmo 8 descreve o *pipeline* de execução. Para garantir a reprodutibilidade, o parâmetro de aleatoriedade (*random*_state) foi definido como 42 e um tamanho de teste definido como 30% do dataset. A Tabela 42 descreve os parâmetros de busca utilizados na otimização de parâmetros. A implementação da experimentação foi disponibilizada publicamente em repositório no Github[3].

Como estudo piloto, o *pipeline* foi inicialmente testado em 5 rodadas com 25 amostras, a fim de verificar seu funcionamento. As adaptações necessárias e eventuais falhas foram corrigidas nessa etapa preliminar. Em seguida, o processo completo foi executado para todos os métodos.

Algorithm 7 *Pipeline* de Experimentação de Métodos de Classificação

```
Define data: X ← features, y ← labels
Define candidate models
Define search spaces for hyperparameters
Select text representation (TF-IDF)
Initialize lists for results and metrics
for each model in models do
    for each iteration r = 1 to n_round do
        if training strategy = Hold-out then
            Split data into train and test
        else if training strategy = cross-validation then
            Define validation folds
        end if
        if hyperparameter tuning = Random or Bayes then
            Perform search over hyperparameter space
            Select best model and parameters
        else
            Train model with Default parameters
        end if
        Generate predictions on the test set
        Store predictions, probabilities, and parameters
    end for
end for
Aggregate results across all iterations
Compute evaluation metrics (accuracy, precision, recall, F1, ROC-AUC)
Return final tables of results and metrics =0
```

Tabela 42 – Espaço de busca de hiperparâmetro dos modelos

Modelo	**Parâmetros**
LogReg	$C \in \{0.01, 0.1, 1, 10\}$
RandomForest	$n\_estimators \in \{100, 200, 500\}$, $max\_depth \in \{\text{None}, 10, 20\}$
SVC	$C \in \{0.1, 1, 10\}$
NaiveBayesMultinomial	$\alpha \in \{0.1, 1, 10\}$, $fit\_prior \in \{\text{True}, \text{False}\}$
KNN	$n\_neighbors \in \{3, 5, 7\}$
GradientBoosting	$n\_estimators \in \{100, 200\}$, $learning\_rate \in \{0.05, 0.1\}$, $max\_depth \in \{3, 6, 10\}$
AdaBoost	$n\_estimators \in \{50, 100, 200\}$, $learning\_rate \in \{0.05, 0.1\}$
XGBoost	$n\_estimators \in \{100, 200\}$, $learning\_rate \in \{0.05, 0.1\}$, $max\_depth \in \{3, 6, 10\}$
LightGBM	$n\_estimators \in \{100, 200\}$, $learning\_rate \in \{0.05, 0.1\}$, $max\_depth \in \{3, 6, 10\}$

[3] https://github.com/k3ybladewielder/guimaraes2026experimental

8.4.2 Execução do Experimento

A execução do experimento consistiu em duas etapas: execução do *pipeline* (8) e avaliação dos resultados.

O *pipeline* foi desenvolvido para executar o pré-processamento de forma sistemática: lower case e tf-idf. Todas as palavras são convertidas para minísculo, e, em seguida por meio de tf-idf, os textos são transformados em vetores contendo o total de palavras, e preenchendo com o valor correspondente da sua importância relativa diante do corpus.

Após isso, os modelos são treinados utilizando 70% da base como base de treinamento e 30% como base de teste. Cada modelo é treinado utilizando os parâmetros de tipo de avaliação (hold-out ou *cross-validation*) e uso de otimização de hiperparâmetros (*Default*, *Random Search* ou *Bayesian Search*) durante 35 rodadas.

O tipo de avaliação hold-out utiliza uma amostra fixa como treinamento e as demais como teste. Já o método *cross-validation* dividiu a base de dados em 5 subconjuntos estratificados, de forma que em cada rodada um subconjunto diferente foi usado como teste e os demais como treinamento. Esse processo garante maior robustez na avaliação do modelo, pois todos os exemplos da base são utilizados tanto para treinamento quanto para teste, em diferentes iterações.

A otimização de hiperparâmetros foi realizada de três formas:

- ***Default***: o modelo é treinado diretamente com seus parâmetros padrão.
- ***Random Search***: amostras aleatórias de combinações de hiperparâmetros são avaliadas para encontrar melhores configurações.
- ***Bayesian Search***: utiliza um processo iterativo baseado em otimização bayesiana, que escolhe novas combinações de parâmetros de forma mais inteligente, levando em conta os resultados anteriores.

Cada rodada gera previsões, probabilidades associadas às classes e registra os parâmetros utilizados, permitindo comparar modelos e estratégias de otimização.

8.4.3 Validação dos Dados

Quatro (4) tipos de testes estatísticos foram empregados para análise, interpretação e validação dos dados: Anderson-Darling (AD Test), Kolmogorov-Smirnov (KS Test), t-test (pareado) e Wilcoxon Signed-Rank Test (pareado). Os testes Anderson-Darling e Kolmogorov-Smirnov foram aplicados para verificar a normalidade dos dados. Para avaliação de modelos pareados que apresentaram evidência de normalidade, utilizou-se o t-test, enquanto o Wilcoxon Signed-Rank Test foi empregado para comparar as medianas das métricas nos casos em que não houve evidência de normalidade.

8.5 Resultados

Nesta seção, são descritos o processo de Análise de Dados e Interpretação, Ameaças à Validade, Conclusões e Trabalhos Futuros.

8.5.1 Análise e Interpretação dos Dados

Para responder às questões de pesquisa presentes em 8.3.4, a etapa de execução foi realizada, e os resultados das classificações foram obtidos para as métricas de avaliação definidas. A Imagem 15 representa visualmente o desempenho por modelo, ordenado pelo F1 Score.

Os modelos SVC (*cross-validation random*), Random Forest (*cross-validation Default*) e Random Forest (*cross-validation random*) apresentaram as melhores acurácias. Enquanto que os modelos Gradient Boosting (*cross-validation Default*), Gradient Boosting (*cross-validation* bayes) e Gradient Boosting (*cross-validation random*) apresentaram os melhores precision. Já sobre o recall, os modelos Random Forest (*cross-validation random*), Random Forest (*cross-validation Default*) e SVC (*cross-validation Default*) se destacaram. Por fim, os modelos Random Forest (*cross-validation Default*), SVC (*cross-validation random*) e Random Forest (*cross-validation random*) apresentaram melhores resultados médios para F1-score. No geral, com exceção da precision, os modelos Random Forest (*cross-validation random*) , Random Forest (*cross-validation Default*) e SVC (*cross-validation random*) destacaram-se sob todas as métricas.

Para compararmos o quão bons os algoritmos são entre si, precisamos ter evidências estatísticas conclusivas. Para isso, os testes Anderson-Darling (AD Test) e Kolmogorov-Smirnov (KS Test) foram executados. Temos evidências de que alguns modelos e configurações seguem uma distribuição normal, conforme Tabela 43.

Os resultados foram divididos em dois grupos, o que temos evidência de que seguem uma distribuição normal (Grupo A) e o grupo que não temos evidência de que os resultados seguem uma distribuição normal (Grupo B). Para avaliar o desempenho dos modelos de forma pareada no Grupo A, utilizou-se o teste de hipótese t-test pareado, enquanto que para o Grupo B, utilizou-se o teste de hipótese Wilcoxon Signed-Rank Test. A Tabela 44 descreve a quantidade de vezes que o modelo avaliado foi melhor do que os demais. A Imagem 16 ilustra o total de pontos por modelo.

Para analisar a eficiência dos algoritmos, focou-se na identificação em ordens de grandeza das operações da contagem de operações básicas do algoritmo como principal indicador da sua eficiência (LEVITIN, 2012). Para comparar e ranquear essas ordens de grandeza, utilizou-se a notação Big-O. A Tabela 45 descreve os modelos selecionados, e sua complexidade para treinamento e predição. As operações básicas foram identificadas com as seguintes notações:

- n = número de amostras do dataset;
- d = número de atributos (dimensionalidade);

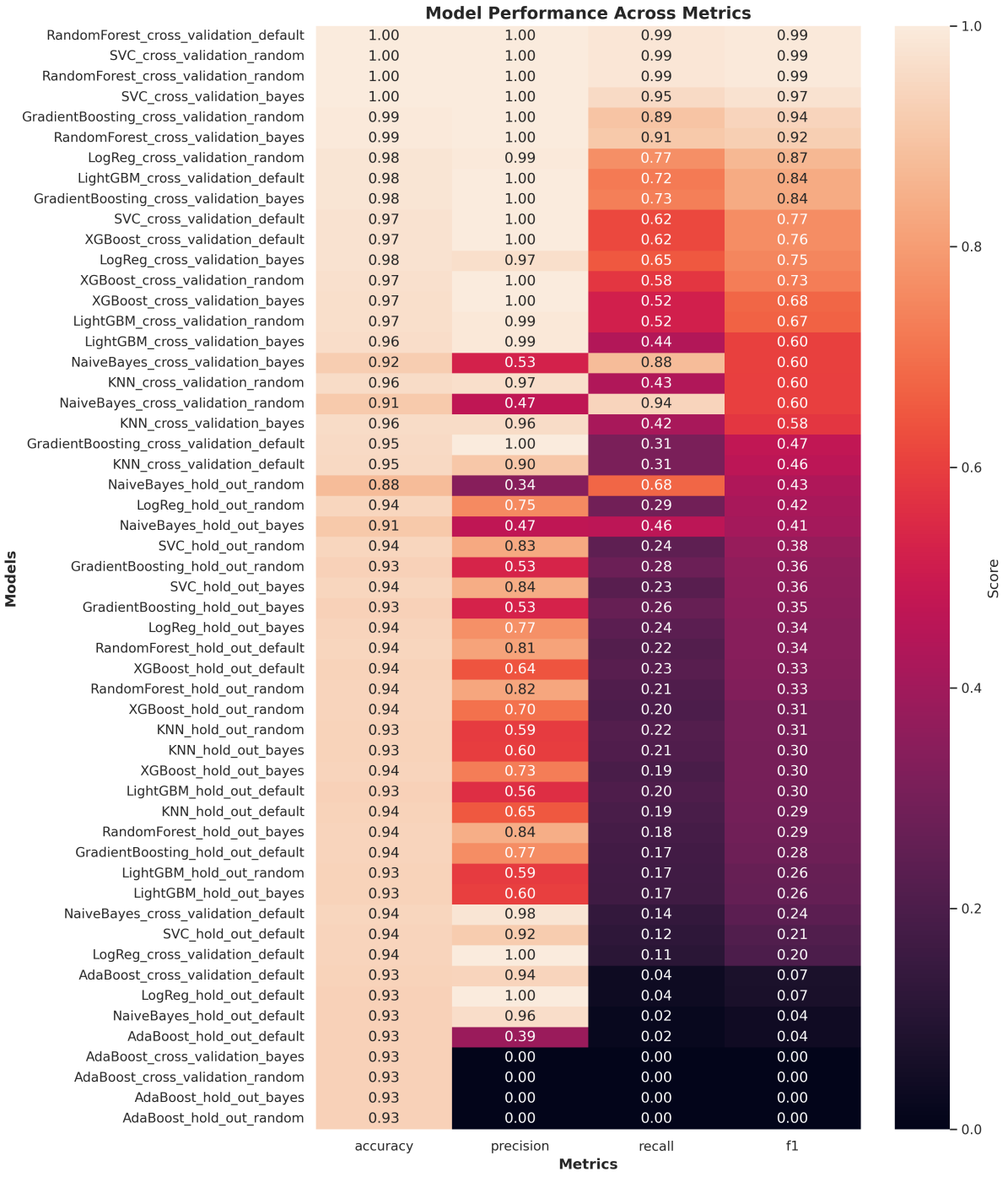


Figura 15 – *Heatmap* das métricas de avaliação por modelo. Ordenado pelo F1 score.

- I = número de iterações do otimizador;
- T = número de árvores (para ensembles e boosting);
- D = profundidade média das árvores;
- C = número de classes;
- s = número de vetores de suporte (no SVC kernelizado).

Tabela 43 – Resultados do teste estatístico AD e rejeição ao nível de 5% para diferentes modelos e métricas. Valor crítico de 0,719 e N de 35.

Model Card	**AD_Statistic**				**AD_Reject_at_5%**			
	accuracy	f1	precision	recall	accuracy	f1	precision	recall
AdaBoost_cross_validation_bayes	0.000000	0.000000	0.000000	0.000000	False	False	False	False
AdaBoost_cross_validation_default	0.000000	0.000000	34.863135	34.863135	False	False	True	True
AdaBoost_cross_validation_random	0.000000	0.000000	0.000000	0.000000	False	False	False	False
AdaBoost_hold_out_bayes	0.199108	0.000000	0.000000	0.000000	False	False	False	False
AdaBoost_hold_out_default	0.298551	1.953894	1.861945	2.102234	False	True	True	True
AdaBoost_hold_out_random	0.199108	0.000000	0.000000	0.000000	False	False	False	False
GradientBoosting_cross_validation_bayes	0.858945	0.810436	0.000000	0.858945	True	True	False	True
GradientBoosting_cross_validation_default	0.444682	0.445671	0.000000	0.444682	False	False	False	False
GradientBoosting_cross_validation_random	4.275143	4.802213	0.000000	4.275143	True	True	False	True
GradientBoosting_hold_out_bayes	0.197463	0.233263	0.319014	0.284222	False	False	False	False
GradientBoosting_hold_out_default	0.554966	0.258988	0.187784	0.280080	False	False	False	False
GradientBoosting_hold_out_random	0.506313	0.153878	0.508408	0.368565	False	False	False	False
KNN_cross_validation_bayes	10.434122	10.434122	10.434122	10.434122	True	True	True	True
KNN_cross_validation_default	34.863135	0.000000	34.863135	34.863135	True	False	True	True
KNN_cross_validation_random	34.863135	34.863135	34.863135	34.863135	True	True	True	True
KNN_hold_out_bayes	0.191272	0.316983	0.420982	0.293760	False	False	False	False
KNN_hold_out_default	0.638736	0.154533	0.320086	0.248038	False	False	False	False
KNN_hold_out_random	0.161830	0.291198	0.572434	0.341261	False	False	False	False
LightGBM_cross_validation_bayes	0.311995	0.305245	2.518436	0.323909	False	False	True	False
LightGBM_cross_validation_default	34.863135	34.863135	34.863135	0.000000	True	True	True	False
LightGBM_cross_validation_random	2.995580	3.060265	8.790676	2.976609	True	True	True	True
LightGBM_hold_out_bayes	0.267097	0.578166	0.586358	0.478697	False	False	False	False
LightGBM_hold_out_default	0.446770	0.373120	0.246842	0.330400	False	False	False	False
LightGBM_hold_out_random	0.246480	0.583085	0.452593	0.474505	False	False	False	False
LogReg_cross_validation_bayes	9.628126	9.503945	12.799091	9.629308	True	True	True	True
LogReg_cross_validation_default	0.000000	34.863135	0.000000	34.863135	False	True	False	True
LogReg_cross_validation_random	34.863135	0.000000	34.863135	0.000000	True	False	True	False
LogReg_hold_out_bayes	0.330636	4.456116	2.654289	3.697330	False	True	True	True
LogReg_hold_out_default	0.193064	0.507881	0.000000	0.536271	False	False	False	False
LogReg_hold_out_random	0.145412	0.666314	0.269488	0.860037	False	False	False	True
NaiveBayes_cross_validation_bayes	7.197509	5.721915	7.722218	5.587578	True	True	True	True
NaiveBayes_cross_validation_default	0.000000	0.000000	34.863135	34.863135	False	False	True	True
NaiveBayes_cross_validation_random	10.434122	10.434122	10.434122	10.434122	True	True	True	True
NaiveBayes_hold_out_bayes	1.926970	2.942268	1.034293	1.929615	True	True	True	True
NaiveBayes_hold_out_default	0.223302	0.366367	11.949051	0.362413	False	False	True	False
NaiveBayes_hold_out_random	3.922155	0.408525	4.972972	4.544995	True	False	True	True
RandomForest_cross_validation_bayes	11.744037	11.745579	7.687040	11.649086	True	True	True	True
RandomForest_cross_validation_default	0.000000	8.634957	8.634957	8.634957	False	True	True	True
RandomForest_cross_validation_random	12.437239	11.606703	6.341815	5.808617	True	True	True	True
RandomForest_hold_out_bayes	0.409875	2.689163	0.619997	2.089862	False	True	False	True
RandomForest_hold_out_default	0.436902	0.310446	0.271578	0.233527	False	False	False	False
RandomForest_hold_out_random	0.342452	2.056600	0.395991	1.408790	False	True	False	True
SVC_cross_validation_bayes	11.781270	11.781270	0.000000	11.781270	True	True	False	True
SVC_cross_validation_default	34.863135	0.000000	0.000000	34.863135	True	False	False	True
SVC_cross_validation_random	0.000000	0.000000	0.000000	34.863135	False	False	False	True
SVC_hold_out_bayes	0.329687	1.043737	0.360275	0.848211	False	True	False	True
SVC_hold_out_default	0.348676	0.376290	0.562678	0.465908	False	False	False	False
SVC_hold_out_random	0.409029	0.187146	0.366474	0.413657	False	False	False	False
XGBoost_cross_validation_bayes	0.526720	0.679656	7.630495	0.523515	False	False	True	False
XGBoost_cross_validation_default	0.000000	34.863135	34.863135	0.000000	False	True	True	False
XGBoost_cross_validation_random	4.071622	3.792327	6.011999	4.077454	True	True	True	True
XGBoost_hold_out_bayes	0.349994	0.160523	0.346832	0.215693	False	False	False	False
XGBoost_hold_out_default	0.246101	0.554307	0.181937	0.425533	False	False	False	False
XGBoost_hold_out_random	0.371639	0.613486	0.389416	0.434931	False	False	False	False

Entre os algoritmos de aprendizado de máquina, aqueles com menor tempo de treinamento são o Naive Bayes e o KNN, ambos com complexidade linear em relação ao número de amostras e atributos. O primeiro realiza apenas a contagem de frequências, enquanto o segundo se limita ao armazenamento dos dados. Em seguida, destaca-se a Logistic Regression, também linear, mas cujo custo depende do número de iterações do otimizador. Métodos de *boosting* mais recentes, como XGBoost, LightGBM e CatBoost, apresentam maior eficiência na construção de árvores em

Tabela 44 – Pontuação total dos 15 melhores modelos por métrica e soma geral.

Modelo	**Accuracy**	**F1**	**Precision**	**Recall**	**ROC AUC**	**Total**
RandomForest_cross_validation_default	51	51	42	49	53	246
RandomForest_cross_validation_random	49	50	40	49	51	239
SVC_cross_validation_random	51	50	45	48	45	239
RandomForest_cross_validation_bayes	49	49	40	47	51	236
SVC_cross_validation_bayes	49	49	45	47	46	236
GradientBoosting_cross_validation_random	48	48	45	46	45	232
SVC_cross_validation_default	43	43	45	40	49	220
LogReg_cross_validation_random	46	46	34	44	48	218
GradientBoosting_cross_validation_bayes	44	44	45	41	43	217
LogReg_cross_validation_bayes	45	45	34	42	45	211
LightGBM_cross_validation_default	44	44	39	41	43	211
XGBoost_cross_validation_default	42	42	38	39	36	197
XGBoost_cross_validation_random	41	41	40	38	36	196
XGBoost_cross_validation_bayes	39	39	40	35	35	188
LightGBM_cross_validation_random	39	39	34	35	35	182
KNN_cross_validation_random	37	36	32	33	39	177
GradientBoosting_cross_validation_default	35	33	45	30	30	173
LightGBM_cross_validation_bayes	37	34	34	32	33	170
KNN_cross_validation_bayes	36	34	31	32	37	170
KNN_cross_validation_default	34	32	28	30	32	156

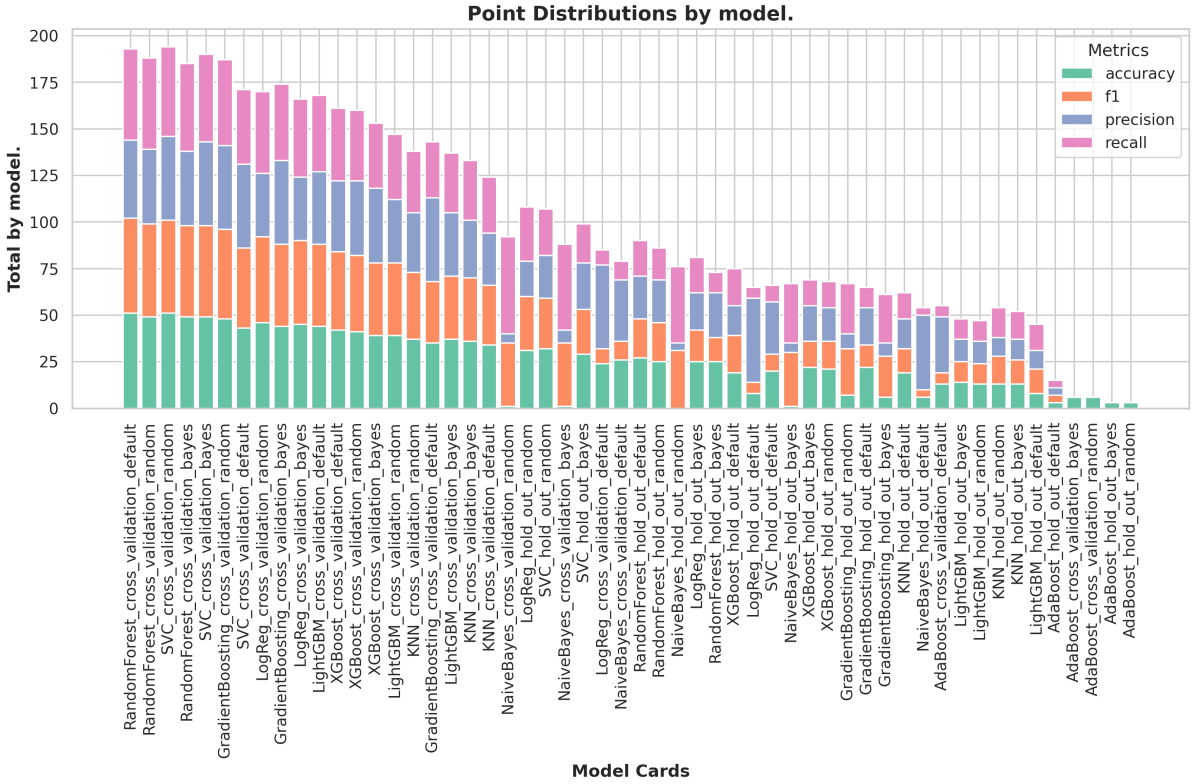


Figura 16 – Distribuição de pontos por modelo.

comparação ao AdaBoost tradicional. Já o Random Forest e o Gradient Boosting do *scikit-learn* demandam maior custo computacional, com complexidade de $O(nd \log n)$ por árvore. Por fim, o C-SVC (libsvm) é o mais oneroso, uma vez que a otimização quadrática cresce entre $O(n^2 d$ e

Tabela 45 – Resumo dos modelos de classificação, complexidades e uso de memória.

Modelo	**Descrição**	**Treinamento**	**Predição**	**Uso de Memória**
Logistic Regression	Maximize probability via logistic regression	$O(n \cdot d \cdot I)$	$O(d)$	$O(d)$
RandomForest	Ensemble de T trees trained on subsets of the dataset	$O(T \cdot n \cdot d \cdot \log n)$	$O(T \cdot D)$	$O(T \cdot n)$
GradientBoosting	Sequential trees correcting errors from previous ones	$O(T \cdot n \cdot d \cdot \log n)$	$O(T \cdot D)$	$O(T \cdot n)$
AdaBoost	T weighted combined weak classifiers	$O(T \cdot n \cdot d)$	$O(T \cdot D)$	$O(T \cdot n)$
C-SVC	Multiclass SVM via one-vs-one with kernel	$O(n^2 \cdot d + n^3)$ per classifier	$O(s \cdot d)$	$O(n^2)$
Naive Bayes (MultinomialNB)	Frequency count by class, assumes independence	$O(n \cdot d)$	$O(d)$	$O(d \cdot C)$
KNeighborsClassifier	Lazy learning, classify by proximity	$O(n \cdot d)$	$O(n \cdot d)$ (ou $O(d \cdot \log n)$ com KD-tree)	$O(n \cdot d)$
XGBClassifier	Optimized boosting with histograms	$O(T \cdot n \cdot d)$	$O(T \cdot D)$	$O(T \cdot n)$
LGBMClassifier	Efficient leaf-wise boosting on large datasets	$O(T \cdot (n + d))$	$O(T \cdot D)$	$O(T \cdot n)$

$O(n^3)$, o que inviabiliza sua aplicação em bases de grande escala.

No processo de predição, os algoritmos mais eficientes são o Naive Bayes e a Logistic Regression, ambos com custo linear em relação ao número de atributos por amostra ($O(d)$). Em seguida, encontram-se os ensembles baseados em árvores AdaBoost, RandomForest, GradientBoosting, XGBoost, LightGBM e CatBoost, cujo custo é proporcional ao número de árvores multiplicado pela profundidade média ($O(TD)$), o que se mostra eficiente em cenários práticos. O C-SVC (libsvm) apresenta desempenho intermediário, pois seu custo depende do número de vetores de suporte ($O(sd)$), podendo variar conforme o conjunto de dados. Já o KNN é o menos eficiente, exigindo a comparação da amostra de teste com todos os pontos armazenados ($O(nd)$), o que se torna inviável em bases volumosas.

Quanto ao consumo de memória, os algoritmos mais leves são o Naive Bayes e a Logistic Regression, que necessitam apenas do armazenamento de frequências ou de um vetor de pesos, com custo $O(dC)$ ou $O(d)$. O KNN apresenta maior demanda, uma vez que requer o armazenamento integral do conjunto de dados ($O(nd)$). Os ensembles (AdaBoost, RandomForest, GradientBoosting, XGBoost, LightGBM e CatBoost) ocupam ainda mais espaço, pois mantêm todas as árvores em memória ($O(Tn)$). O maior custo, entretanto, é observado no C-SVC (libsvm), que necessita armazenar a matriz de kernel de dimensão $n \times n$, resultando em custo quadrático ($O(n^2)$), fator que limita sua aplicação em grandes bases de dados.

Sob a perspectiva do auditor, a escolha dos modelos mais adequados depende de fatores como factualidade, urgência, disponibilidade de recursos computacionais, tempo de inferência e tipo de aplicação (fluxo contínuo ou processamento em lote). Em cenários que exigem maior precisão quanto à identificação de notícias com indícios de irregularidade, avaliadas por métricas como *accuracy*, *precision*, *recall* e *F1-score*, é preferível a utilização de modelos que apresentem melhores combinações dessas métricas, como *Random Forest* (*cross-validation Default*), *Random*

Forest (*cross-validation random*) e *SVC* (*cross-validation random*), que se destacaram sob todas as métricas de avaliação do experimento. Em situações que demandam maior urgência e agilidade no planejamento, recomenda-se a utilização de modelos com menor complexidade teórica de treinamento, predição e uso de memória, como Naive Bayes, Logistic Regression ou *LightGBM*.

8.6 Considerações Finais

O processo de auditoria caracteriza-se, em geral, por ser custoso, demorado e demandar substanciais recursos humanos e materiais. Nesse sentido, torna-se necessário implementar soluções e técnicas que possibilitem a automatização da análise de denúncias de corrupção. Esse processo é usualmente dividido em duas etapas: na primeira, busca-se identificar elementos e evidências de corrupção, tais como fornecedores, contratos, funcionários, clientes e demais partes interessadas, avaliando a plausibilidade e consistência das denúncias e indícios de fraude; na segunda etapa, desenvolve-se a investigação propriamente dita.

Para a construção do conhecimento necessário à atividade de auditoria, é imprescindível o levantamento de informações relacionadas ao objetivo da auditoria. Nessa fase, recorre-se a diversas fontes, incluindo páginas da internet. Para apoiar o processo de coleta de informações, podem ser aplicadas técnicas de *web scraping* voltadas à extração massiva de dados em sites do contexto da saúde. Ademais, para auxiliar na análise dessa grande quantidade de dados, técnicas de PLN, como a sumarização de texto, podem ser empregadas, reduzindo significativamente o tempo e os recursos necessários para a análise e a coleta de evidências de possíveis irregularidades.

Neste contexto, com o objetivo de apoiar, aprimorar e otimizar a coleta de informações relevantes que possam auxiliar no combate a irregularidades, este trabalho apresenta os resultados da aplicação de 54 modelos de classificação de machine learning aplicados em um conjunto de notícias relacionadas à área da saúde com indícios de irregularidade. Buscou-se, assim, avaliar se tais métodos podem contribuir para o processo de auditoria, identificando diretamente quais notícias possuem indícios de irregularidade, bem como identificar quais se mostram mais eficientes e eficazes para essa tarefa.

Neste experimento controlado **in vitro**, utilizando uma base de dados curada com 6239 amostras de notícias, foram executados 54 modelos de classificação de machine learning, repetidos em 35 rodadas para mensurar a eficácia e eficiência dos modelos de forma robusta. Para analisar eficácia, utilizou-se as métricas de desempenho Acurácia, Precision, Recall e F1-Score (ZHU; ZENG; WANG, 2010). Para mensurar a eficiência, utilizou-se análise assintótica (Big-O) para identificar a complexidade de treinamento, predição e uso de memória.

Quanto a eficácia, os modelos SVC (*cross-validation random*), Random Forest (*cross-validation Default*) e Random Forest (*cross-validation random*) apresentaram as melhores acurácias. Enquanto que os modelos Gradient Boosting (*cross-validation Default*), Gradient Boosting (*cross-validation* bayes) e Gradient Boosting (*cross-validation random*) apresentaram

os melhores precision. Já sobre o recall, os modelos Random Forest (*cross-validation random*), Random Forest (*cross-validation Default*) e SVC (*cross-validation Default*) se destacaram. Por fim, os modelos Random Forest (*cross-validation Default*), SVC (*cross-validation random*) e Random Forest (*cross-validation random*) apresentaram melhores resultados médios para F1-score. No geral, com exceção da precision, os modelos Random Forest (*cross-validation random*) , Random Forest (*cross-validation Default*) e SVC (*cross-validation random*) destacaram-se sob todas as métricas.

Com os resultados avaliados sob estas métricas, analisou-se a hipótese de normalidade por meio dos testes Anderson-Darling (AD) e Kolmogorov-Smirnov (KS). Na análise pareada, os resultados que apresentaram evidência de normalidade utilizam-se do t-test, enquanto o Wilcoxon Signed-Rank Test foi empregado para comparar as medianas das métricas nos casos em que não houve evidência de normalidade, cujos resultados são apresentados neste estudo. Nos testes pareados, os modelos Random Forest (*cross-validation random*), Random Forest (*cross-validation Default*) e SVC (*cross-validation random*) se destacaram como o primeiro, segundo e terceiro melhores dentre todos.

Quanto a eficiência, entre os algoritmos de aprendizado de máquina, aqueles com menor tempo de treinamento são o Naive Bayes e o KNN, ambos com complexidade linear em relação ao número de amostras e atributos. No processo de predição, os algoritmos mais eficientes são o Naive Bayes e a Logistic Regression, ambos com custo linear em relação ao número de atributos por amostra ($O(d)$). Quanto ao consumo de memória, os algoritmos mais leves são o Naive Bayes e a Logistic Regression, que necessitam apenas do armazenamento de frequências ou de um vetor de pesos, com custo $O(dC)$ ou $O(d)$.

Em conclusão, sob a perspectiva do auditor, a escolha dos modelos mais adequados depende de fatores como factualidade, urgência, disponibilidade de recursos computacionais, tempo de inferência e tipo de aplicação (fluxo contínuo ou processamento em lote). Em cenários que exigem maior precisão quanto à identificação de notícias com indícios de irregularidade, avaliadas por métricas como *accuracy*, *precision*, *recall* e *F1-score*, é preferível a utilização de modelos que apresentem melhores combinações dessas métricas, como *Random Forest* (*cross-validation Default*), *Random Forest* (*cross-validation random*) e *SVC* (*cross-validation random*), que se destacaram sob todas métricas de avaliação do experimento. Em situações que demandam maior urgência e agilidade no planejamento, recomenda-se a utilização de modelos com menor complexidade teórica de treinamento, predição e uso de memória, como Naive Bayes, Logistic Regression ou *LightGBM*.

A seguir, buscando avaliar métodos que auxiliem a tarefa de resumo de texto para otimizar e maximizar a redução de sobrecarga informacional e o processo de auditoria, no Capítulo 9, a avaliação experimental de métodos de modelagem de tópicos é apresentada.

9

Avaliação Experimental de Métodos de Modelagem de Tópicos

Este capítulo apresenta parcialmente experimento controlado intitulado *Experimental Evaluation of Topic Modeling Methods for Categorizing Irregularities in Health-related news* à conferência 7th International Conference on Computational Processing of Portuguese (PROPOR 2026).

9.1 Contextualização

Este estudo tem como objetivo analisar métodos de modelagem de tópicos, sob a perspectiva de Cientistas de Dados e Auditores do Sistema Único de Saúde (SUS), no contexto de auditorias em saúde pública conduzidas pelo Departamento Nacional de Auditoria do SUS (AudSUS).

Neste estudo, conduzimos um experimento controlado *in vitro* para avaliar o desempenho dos métodos em tarefas de modelagem de tópicos, utilizando as métricas de coerência CV e $CNPMI$. Adicionalmente, foram avaliadas a consistência dos resultados ao longo de 35 execuções e as performances obtidas.

O método LSA destacou-se entre os modelos com as maiores médias de coerência CV e $CNPMI$, seguido por NMF e pLSA para CV, e por LSA e NMF para $CNPMI$, respectivamente. Modelos baseados em LSA apresentaram desempenho superior em comparação aos demais 215 modelos avaliados, especialmente em configurações com valores reduzidos de $top\text{-}n$ e $top\text{-}k$. As variantes de LSA com $top\text{-}n = 5$ alcançaram as maiores pontuações em ambas as métricas de coerência, CV e $CNPMI$, ocupando as primeiras posições no *ranking* geral. A configuração com $top\text{-}n = 5$ e $top\text{-}k = 5$ mostrou-se a mais eficaz, superando todos os outros 215 métodos avaliados. De modo geral, a análise estatística confirma que as diferenças observadas entre os modelos não se devem à variação aleatória.

Os resultados evidenciam o elevado potencial dos métodos de modelagem de tópicos para o agrupamento de notícias que apresentam indícios de irregularidades, orientando a recuperação de informações durante a fase analítica do processo de auditoria. Essa abordagem aprimora a efetividade global das auditorias e facilita a preparação mais ágil das equipes para a etapa operacional.

9.2 Materiais e Métodos

Este é um estudo experimental, seguindo os passos apresentados por Colaço JÚNIOR et al. (2022), Colaço JÚNIOR (2025) para avaliar os resultados de modelos de modelagem de tópicos aplicados, utilizando notícias relacionadas à saúde com indícios de irregularidade, e avaliando a coerência interna por meio das métricas Valor de Coerência (CV) e Informação Mútua Pontual Normalizada ($CNMPI$). O processo experimental foi descrito na Seção 2.9.1.

A descrição do banco de dados utilizado, o processo de seleção de notícias com indícios de irregularidade e as métricas de avaliação dos resultados são descritos na Subsessão 2.9.1 e 2.7.3, respectivamente.

9.3 Configuração Experimental

Nesta seção, são apresentados o objetivo da avaliação experimental, o planejamento, as perguntas de pesquisa, as variáveis independentes, as variáveis dependentes e as hipóteses.

9.3.1 Objetivo

Para formalizar o objetivo deste estudo, adotou-se o modelo *Goal Question Metric* (GQM) proposto por Basili e Weiss (1984). Este estudo visa **analisar** métodos de modelagem de tópicos por meio de um experimento controlado (*in vitro*), **com o propósito de** avaliá-los quantitativamente, **em relação às** métricas de CV e $CNPMI$, **em relação aos** temas (tópicos) de notícias relacionadas à saúde com indícios de irregularidade, **sob a perspectiva de** Cientistas de Dados e Auditores do Sistema Único de Saúde (SUS) brasileiro, **no contexto de** auditorias de saúde pública conduzidas pela Secretaria Nacional de Auditoria do SUS (AudSUS).

9.3.2 Planejamento

Neste estudo, o experimento controlado foi orientado à um ambiente de experimento controlado *in vitro*, utilizando a base de dados supracitada na Subseção 2.9.1. O dataset filtrado contêm 421 notícias.

9.3.3 Seleção de Contexto

Apesar dos avanços tecnológicos significativos, diversos procedimentos no setor público ainda dependem da recuperação manual de informações para a construção do conhecimento. Essa realidade também se observa no Departamento Nacional de Auditoria do Sistema Único de Saúde (AudSUS), responsável pela supervisão e auditoria das operações do SUS. As atividades de auditoria conduzidas por essa instituição são fundamentais para assegurar a gestão eficiente e a adequada alocação de recursos públicos; contudo, o processo permanece altamente demandante em termos de recursos, em razão da elevada carga de trabalho, uma vez que os auditores devem acompanhar todos os domínios do SUS e, simultaneamente, atender a demandas internas e externas do Ministério da Saúde (FONTES et al., 2023). Nesse contexto, o experimento proposto busca apoiar a fase analítica do processo de auditoria — responsável pelo planejamento e pela preparação da equipe para a fase operacional — por meio da coleta de informações relevantes aos objetivos da auditoria.

9.3.4 Questões de Pesquisa

Para orientar o experimento e atingir o objetivo do estudo, foram formuladas as seguintes questões de pesquisa:

- RQ1: Qual dos métodos selecionados é o melhor em termos de coerência?
- RQ2: Dentre os métodos selecionados, quais são os mais consistentes em termos de coerência?

Para abordar as questões de pesquisa, foram criadas as seguintes hipóteses teóricas descritas na Tabela 46.

Tabela 46 – Questões de pesquisa e hipóteses associadas

RQ	**Hipótese Nula (H_0)**	**Hipótese Alternativa (H_1)**
RQ1	Os métodos não apresentam diferenças significativas em termos de coerência.	Os métodos apresentam diferenças significativas em termos de coerência.
RQ2	Os métodos não são consistentes ao longo das rodadas.	Os métodos são consistentes ao longo das rodadas.

9.3.5 Variáveis Dependentes

As variáveis dependentes, ou variáveis de saída, foram os artigos de notícias categorizados, a partir dos quais as métricas CV e $CNPMI$ podem ser derivadas.

9.3.6 Variáveis Independentes

Neste experimento, as variáveis independentes, ou variáveis de entrada, são: o conjunto de dados anotado (classificado) de artigos de notícias com indícios de irregularidade; e os métodos utilizados para a tarefa de modelagem de tópicos: BERTopic, HDP, LDA, LSA, NMF e pLSA.

9.3.7 Objects Selection

Conforme o contexto descrito em 9.3.3, os objetos deste experimento consistem em artigos de notícias relacionados à saúde que indicam potenciais irregularidades, conforme descrito na Seção 2.9.1. Para a determinação do tamanho da amostra, considerou-se uma população finita de 154.407 artigos de notícias — representando o número total de itens no conjunto de dados completo. É importante notar que a amostra final ultrapassa o tamanho estimado calculado de acordo com a Eq. 9.2. A estimativa da amostra foi baseada em um nível de confiança de 95% ($Z = 1,96$), um erro amostral tolerável de 5% ($e = 0,05$) e uma proporção esperada de 50% ($p = 0,5$), parâmetros que maximizam a variabilidade da amostra e garantem uma estimativa conservadora do tamanho da amostra.

O tamanho da amostra para uma população finita foi calculado em duas etapas: inicialmente, o tamanho da amostra para uma população infinita (n) foi estimado usando a Eq. 9.1, seguido por um ajuste para uma população finita ($n_{ajustado}$) de acordo com a Eq. 9.2, resultando em aproximadamente 383,21 amostras, conforme mostrado na Eq. 9.4. Por fim, todas as amostras classificadas manualmente do conjunto de dados foram utilizadas, sendo 421 classificadas como "Irregularidade em Saúde", excedendo assim o requisito mínimo de 384 artigos para uma amostra representativa.

$$n = \frac{Z^2 \cdot p \cdot (1-p)}{e^2} \tag{9.1}$$

$$n_{adjusted} = \frac{n}{1 + \left(\frac{n-1}{N}\right)} \tag{9.2}$$

$$n = \frac{1{,}96^2 \cdot 0{,}5 \cdot (1-0{,}5)}{0{,}05^2} = \frac{3{,}8416 \cdot 0{,}25}{0{,}0025} = 384{,}16 \tag{9.3}$$

$$n_{adjusted} = \frac{384{,}16}{1 + \left(\frac{384{,}16-1}{154407}\right)} \approx 383{,}21 \tag{9.4}$$

9.3.8 Configuração do Experimento

A execução foi projetada para seguir um processo sistemático envolvendo etapas de pré-processamento, como conversão para minúsculas, remoção de stopwords e stemming (snowball

stemming) em português brasileiro, implementado com a biblioteca NLTK do Python. Em seguida, o processo TF-IDF visa capturar a importância relativa dos termos nas manchetes das notícias e gerar os atributos (características) ou variáveis independentes do conjunto de dados, conforme descrito por Salton e Buckley (1988), utilizando a biblioteca Scikit-learn do Python. Após a fase de preparação dos dados, o pipeline de modelagem de tópicos foi aplicado.

9.3.9 Instrumentação

Os seguintes materiais e recursos foram utilizados:

- Conjunto de dados anotado com resumos de referência (2.9.1);
- Linguagem de programação Python (3.11.13)[1];
- Bibliotecas Python: bertopic (0.17.3), gensim (4.3.3), ipykernel (7.0.0), matplotlib (3.10.5), nltk (3.9.2), numpy (2.0.0), openpyxl (3.1.4), pandas (2.3.1), polars (1.34.0), pyarrow (21.0.0), scikit-learn (1.7.1), seaborn (0.13.2), tqdm (4.67.1) e uv (0.8.14); Computador com processador Intel® Core™ i5-1235Ux12 de 12ª geração e 16 GB de RAM; Recursos computacionais do Núcleo de Computação de Alto Desempenho (NPAD) da Universidade Federal do Rio Grande do Norte (UFRN).

9.4 Operacionalização do Experimento

Esta seção descreve o processo de preparação do experimento, sua execução e avaliação dos resultados. A Figura 17 ilustra a preparação e a execução do experimento realizadas nas subseções seguintes.

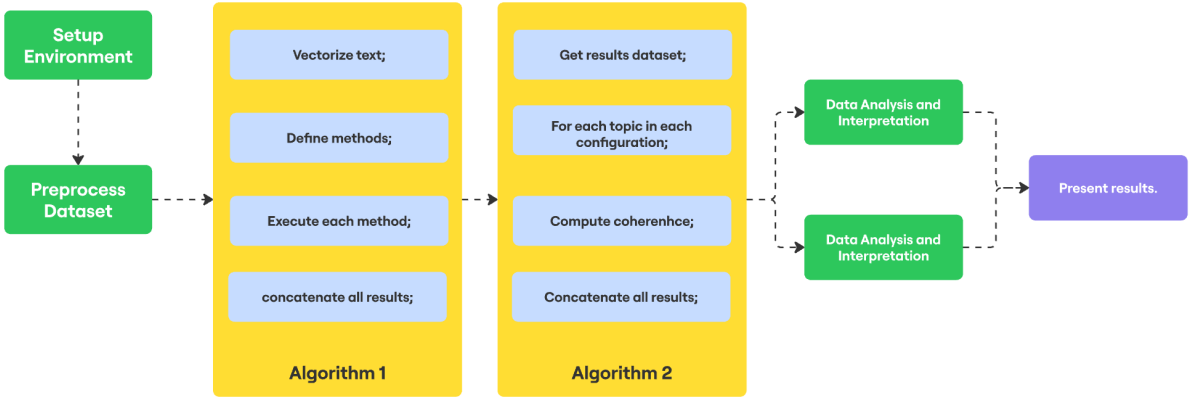


Figura 17 – Preparação e Execução do Experimento.

[1] https://www.python.org/

9.4.1 Preparação do Experimento

O ambiente de execução, tanto localmente quanto no supercomputador NPAD, foi configurado utilizando a biblioteca *uv*, que foi empregada para criar um ambiente virtual e instalar as bibliotecas necessárias descritas na Subseção 9.3.9 para a tarefa de modelagem de tópicos. Posteriormente, o conjunto de dados foi utilizado dentro desse ambiente. Para garantir um processo de execução sistemático e reproduzível, um *pipeline* de modelagem de tópicos foi desenvolvido. Este *pipeline* compreende um script que executa os procedimentos para cada método candidato descrito na Subseção 9.3.6 e para cada parâmetro utilizado, como top-n (palavras mais frequentes) e top-k (número de tópicos). Seis valores foram considerados para top-n e top-k: 5, 10, 15, 20, 25 e 30. Combinando todas as configurações de parâmetros por meio de uma busca em grade, um total de 216 modelos (configurações) foram gerados para os parâmetros método, top-n e top-k. O algoritmo 8 apresenta os detalhes do pipeline de execução. Para garantir a reprodutibilidade, o parâmetro aleatório (`random_state`) foi atribuído para corresponder à rodada de teste atual, variando de 1 a 35 para cada configuração ao longo das 35 rodadas. O parâmetro de aleatoriedade é diferente para cada rodada, a fim de gerar resultados diferentes a partir de inicializações diferentes, com o objetivo de avaliar a consistência dos resultados sob diferentes condições, conforme as Questões de Pesquisa em 9.3.4, garantindo a robustez dos resultados.

Como um estudo piloto, o *pipeline* foi inicialmente testado em 5 rodadas com 5 amostras para verificar seu funcionamento. Ajustes necessários e possíveis falhas foram corrigidos durante esta etapa preliminar. Posteriormente, o processo completo foi executado para todos os métodos.

9.4.2 Execução do Experimento

O experimento foi conduzido em duas etapas: a execução do pipeline de modelagem de tópicos e a avaliação dos resultados.

O pipeline para modelagem de tópicos foi projetado para realizar o pré-processamento de forma sistemática, incluindo a conversão para minúsculas e a vetorização TF-IDF. Todas as palavras foram convertidas para minúsculas e, posteriormente, por meio da transformação TF-IDF, os textos foram convertidos em vetores representando o vocabulário total, com cada elemento ponderado de acordo com sua importância relativa dentro do corpus. Em seguida, para cada método avaliado, um loop de geração de tópicos foi executado, e as saídas resultantes foram armazenadas e concatenadas em um único dataframe contendo todos os tópicos gerados por todos os métodos. Todo o pipeline de modelagem de tópicos é descrito no Algoritmo 8.

Posteriormente, um pipeline de avaliação empregando as métricas CV e $CNPMI$ foi executado, produzindo as pontuações correspondentes para cada tópico em cada rodada de execução e método, conforme descrito no Algoritmo 9. Finalmente, os dados resultantes foram armazenados para análise posterior.

A implementação de ambos os *pipelines* está disponível publicamente em repositório no Github[2].

Algorithm 8 *Pipeline* de Experimentação de Modelagem de Tópicos

Input: Dataset df, column name $text\_col$, number of topics n_topics, number of rounds n_rounds
Extract texts: $texts \leftarrow df[text\_col]$ (fill missing, convert to string)
Initialize **vectorizer** ← TF-IDF(max_features = 5000)
Compute document-term matrix $X \leftarrow vectorizer.fit\_transform(texts)$
Initialize empty list $results \leftarrow []$
Define **methods** ← [NMF, LSA, LDA, BERTopic, HDP, pLSA]
for each **method** in methods **do**
  for each round $i = 1$ **to** n_rounds **do**
    Train $method(texts)$
    $labels \leftarrow topics$
    Create temporary dataframe $df\_tmp \leftarrow df.copy()$
    Add columns: method, round = i, topic = $labels$
    Append df_tmp to $results$ exception e
    Print error message $[ERRO]\ method, round, e$
  end for
end for
Concatenate all partial results $df\_final \leftarrow concat(results)$
Return df_final =0

Algorithm 9 *Pipeline* de Avaluação de Coerência de Tópicos

Input: Dataset df_topics, columns $(text\_col, topic\_col, method\_col, round\_col)$, metrics set $metrics = \{c\_v, c\_npmi\}$, and top-$n$ words $topn$
Tokenize texts: $df\_topics[\_tokens\_] \leftarrow split(df\_topics[text\_col],)$
Initialize empty list $results \leftarrow []$
Partition dataset by $(method\_col, round\_col)$ into groups $grouped$
for each $configuration$ **do**
  Extract tokenized documents: $tokenized\_docs \leftarrow group\_df[\_tokens\_]$
  Build dictionary: $dictionary \leftarrow Dictionary(tokenized\_docs)$
  Build corpus:
  $corpus \leftarrow [dictionary.doc2bow(doc)$ for each $doc \in tokenized\_docs]$
  Initialize empty list $topics \leftarrow []$
  for each $metric$ in $metrics$ **do**
    Compute coherence model
    $cm \leftarrow CoherenceModel(topics, tokenized\_docs, dictionary, metric)$
    Get coherence score $coh \leftarrow cm.get\_coherence()$
    Append result $\{method, rnd, metric, coh, |topics|, |group\_df|, top-n\}$ to $results$
  end for
  exception e
  Print error message $[ERROR]\ method, rnd, e$
end for
Return $DataFrame(results)$ =0

9.4.3 Data Validation

Quatro (4) testes estatísticos foram empregados para a análise, interpretação e validação dos dados: o teste de Anderson-Darling (AD), o teste de Kolmogorov-Smirnov (KS), o teste t pareado e o teste de Wilcoxon com sinais ordenados. Os testes de Anderson-Darling e Kolmogorov-Smirnov foram aplicados para avaliar a normalidade dos dados. Para avaliações de modelos pareados que apresentaram evidências de normalidade, utilizou-se o teste t pareado, enquanto o

[2]https://github.com/k3ybladewielder/propor26

teste de Wilcoxon com sinais ordenados foi empregado para comparar os valores medianos das métricas nos casos em que não houve evidências de normalidade.

9.5 Resultados

Esta seção descreve a análise e interpretação dos dados, bem como o processo de avaliação estatística.

9.5.1 Análise e Interpretação dos Dados

Para responder às questões de pesquisa apresentadas em 9.3.4, a etapa de execução foi realizada e as métricas dos tópicos foram obtidas para as métricas de avaliação definidas.

Após combinar os seis métodos com os seis valores top-n e top-k possíveis, um total de 216 modelos (configurações) foram gerados. Cada configuração foi executada em 35 rodadas e os resultados correspondentes foram registrados para calcular a média, a mediana, o mínimo, o máximo e o desvio padrão das métricas de coerência. As tabelas 47 e 48 apresentam os resultados para os 25 modelos com os maiores valores médios de coerência.

Tabela 47 – Estatísticas de coerência para os métodos de modelagem de tópicos usando a métrica *CV*. Ordenadas por coerência média.

Métrica	Config	Média	Mediana	Min	Max	Desvio Padrão	Class
CV	model: LSA, top-n: 5, top-k: 5	0.678336	0.667915	0.667915	0.719436	0.020852	Low
CV	model: LSA, top-n: 5, top-k: 15	0.659714	0.659786	0.647247	0.681399	0.008474	Low
CV	model: LSA, top-n: 5, top-k: 20	0.649771	0.650636	0.627398	0.671620	0.010073	Low
CV	model: LSA, top-n: 5, top-k: 10	0.642928	0.643777	0.588675	0.705938	0.028154	Low
CV	model: LSA, top-n: 5, top-k: 25	0.641559	0.644809	0.611926	0.662006	0.012006	Low
CV	model: LSA, top-n: 5, top-k: 30	0.637768	0.637706	0.614423	0.656624	0.010692	Low
CV	model: NMF, top-n: 5, top-k: 25	0.628399	0.629427	0.615936	0.639792	0.007176	Low
CV	model: NMF, top-n: 5, top-k: 30	0.627461	0.627654	0.613681	0.637752	0.005258	Low
CV	model: pLSA, top-n: 5, top-k: 30	0.623795	0.624451	0.599529	0.639714	0.010080	Low
CV	model: pLSA, top-n: 5, top-k: 25	0.618417	0.616701	0.604708	0.635379	0.007992	Low
CV	model: NMF, top-n: 5, top-k: 20	0.616952	0.616099	0.607562	0.631391	0.007462	Low
CV	model: LDA, top-n: 5, top-k: 25	0.615484	0.611183	0.528175	0.691355	0.040661	Low
CV	model: pLSA, top-n: 5, top-k: 20	0.615416	0.614852	0.589531	0.643414	0.009674	Low
CV	model: LDA, top-n: 5, top-k: 30	0.613711	0.607744	0.556950	0.660186	0.027323	Low
CV	model: pLSA, top-n: 5, top-k: 15	0.610555	0.609609	0.606606	0.617131	0.002584	Low
CV	model: LDA, top-n: 5, top-k: 20	0.608888	0.603238	0.509430	0.724870	0.044335	Low
CV	model: NMF, top-n: 5, top-k: 15	0.608155	0.610050	0.604062	0.610500	0.002786	Low
CV	model: NMF, top-n: 5, top-k: 10	0.603515	0.599745	0.587697	0.615350	0.008095	Low
CV	model: HDP, top-n: 5, top-k: 10	0.592559	0.591658	0.577481	0.610857	0.008853	Low
CV	model: HDP, top-n: 5, top-k: 20	0.592289	0.593295	0.578896	0.607092	0.006418	Low
CV	model: HDP, top-n: 5, top-k: 15	0.591479	0.590582	0.572825	0.615866	0.009491	Low
CV	model: HDP, top-n: 5, top-k: 5	0.591144	0.591471	0.568160	0.604279	0.007333	Low
CV	model: HDP, top-n: 5, top-k: 25	0.590509	0.591438	0.572336	0.608455	0.008185	Low
CV	model: HDP, top-n: 5, top-k: 30	0.589504	0.590268	0.577942	0.602276	0.006206	Low
CV	model: LDA, top-n: 5, top-k: 15	0.586424	0.607397	0.435750	0.702871	0.070912	Moderate

O método LSA destacou-se entre os 10 modelos com a maior coerência média *CV* nas 6 primeiras posições, seguido por NMF e pLSA. Na métrica média $CNPMI$, o método LSA

Tabela 48 – Estatísticas de coerência para modelos de tópicos usando a métrica $CNPMI$. Ordenadas por coerência média.

Métrica	Config	Média	Mediana	Min	Max	Desvio Padrão	Class
$CNPMI$	model: LSA, top-n: 5, top-k: 5	0.175260	0.158527	0.158527	0.231728	0.029230	Low
$CNPMI$	model: LSA, top-n: 5, top-k: 15	0.154070	0.153015	0.134467	0.183592	0.012860	Low
$CNPMI$	model: LSA, top-n: 5, top-k: 10	0.142514	0.143693	0.086927	0.217466	0.037734	Low
$CNPMI$	model: LSA, top-n: 5, top-k: 20	0.141044	0.143768	0.116590	0.165643	0.012833	Low
$CNPMI$	model: LSA, top-n: 5, top-k: 25	0.130320	0.133465	0.092675	0.156592	0.014876	Low
$CNPMI$	model: LSA, top-n: 5, top-k: 30	0.128705	0.129010	0.083141	0.155468	0.015897	Low
$CNPMI$	model: LDA, top-n: 5, top-k: 30	0.115689	0.107268	0.049664	0.201930	0.040176	Low
$CNPMI$	model: LDA, top-n: 5, top-k: 25	0.112683	0.113851	-0.023658	0.236730	0.054868	Moderate
$CNPMI$	model: LSA, top-n: 10, top-k: 5	0.110963	0.111709	0.089543	0.117318	0.006333	Low
$CNPMI$	model: NMF, top-n: 5, top-k: 30	0.108545	0.108976	0.090646	0.121334	0.007590	Low
$CNPMI$	model: NMF, top-n: 5, top-k: 25	0.106101	0.106093	0.087666	0.119363	0.008655	Low
$CNPMI$	model: LDA, top-n: 5, top-k: 20	0.103294	0.096390	-0.011306	0.225153	0.055446	Moderate
$CNPMI$	model: LSA, top-n: 10, top-k: 15	0.099685	0.097103	0.083527	0.133795	0.009520	Low
$CNPMI$	model: pLSA, top-n: 5, top-k: 30	0.095376	0.095939	0.064465	0.120409	0.013760	Low
$CNPMI$	model: LSA, top-n: 15, top-k: 5	0.089554	0.091990	0.073086	0.091990	0.006025	Low
$CNPMI$	model: NMF, top-n: 5, top-k: 20	0.089488	0.086704	0.078364	0.107788	0.009461	Low
$CNPMI$	model: LSA, top-n: 10, top-k: 10	0.089179	0.086933	0.066025	0.117896	0.014617	Low
$CNPMI$	model: LSA, top-n: 10, top-k: 20	0.088942	0.088487	0.072590	0.103280	0.007923	Low
$CNPMI$	model: pLSA, top-n: 5, top-k: 15	0.087932	0.088324	0.080365	0.092949	0.002640	Low
$CNPMI$	model: pLSA, top-n: 5, top-k: 25	0.086617	0.086400	0.066838	0.104188	0.009831	Low
$CNPMI$	model: pLSA, top-n: 5, top-k: 20	0.086184	0.086134	0.064711	0.126687	0.012109	Low
$CNPMI$	model: NMF, top-n: 5, top-k: 15	0.083931	0.083861	0.082803	0.087785	0.001306	Low
$CNPMI$	model: LDA, top-n: 5, top-k: 15	0.083857	0.096668	-0.047820	0.217287	0.086558	Moderate
$CNPMI$	model: LSA, top-n: 10, top-k: 25	0.079554	0.078998	0.060556	0.106252	0.010100	Low
$CNPMI$	model: LSA, top-n: 10, top-k: 30	0.077481	0.079306	0.052841	0.094671	0.009399	Low

também se destacou, ocupando 8 das 10 primeiras posições, seguido por LSA e NMF. Para avaliar a consistência dos resultados, o desvio padrão foi analisado tanto coletivamente quanto visualmente ao longo das rodadas. Para fins comparativos, visto que todas as métricas variam de -1 a 1, os resultados foram classificados com base no valor do desvio padrão para cada métrica como "Baixo"(desvio padrão < 0,05), "Moderado"(desvio padrão > 0,05 e < 0,1) e "Alto"(desvio padrão > 0,1). Dessa perspectiva, entre os cinquenta modelos (configurações de modelo, top-n e top-k) com os maiores valores médios de CV, apenas três apresentaram desvio padrão moderado — a saber, $model : LDA, top-n : 5, top-k : 15$; $model : LDA, top-n : 5, top-k : 10$; e $model : LDA, top-n : 5, top-k : 5$, enquanto os demais modelos apresentaram valores baixos. Para a métrica $CNPMI$, quatro casos mostraram desvio padrão moderado e quarenta e seis apresentaram baixa variabilidade: $model : LDA, top-n : 5, top-k : 25$; $model : LDA, top-n : 5, top-k : 20$; $model : LDA, top-n : 5, top-k : 15$; e $model : LDA, top-n : 10, top-k : 15$. Um baixo desvio padrão, combinado com altos valores de coerência CV e $CNPMI$, indica que os métodos são robustos a variações de parâmetros, mantendo um alto desempenho. As figuras 18 e 19 ilustram esses resultados para os 25 modelos com os maiores índices de coerência CV e $CNPMI$.

Entre os cinquenta modelos com os maiores valores médios de CV, o método LSA destacou-se, aparecendo 20 vezes, enquanto cada um dos outros métodos apareceu seis vezes. Da mesma forma, entre os cinquenta modelos com os maiores valores médios de $CNPMI$, o LSA

foi novamente o mais frequente, com 19 ocorrências, seguido por NMF e pLSA com nove cada, LDA com sete e HDP com seis.

Em relação ao parâmetro $top-n$, os melhores resultados foram obtidos usando apenas as cinco palavras mais frequentes, que ocorreram 36 vezes. O segundo melhor valor de $top-n$ foi 10 (seis ocorrências), seguido por 15 (cinco ocorrências), enquanto os valores restantes apareceram apenas uma vez na métrica CV. Para a métrica $CNPMI$, os melhores resultados também corresponderam a top-n = 5 (25 ocorrências), seguido por $top-n = 10$ (17 ocorrências), 15 (cinco ocorrências) e os valores restantes ocorrendo uma vez cada. De forma semelhante, para a métrica CV, os valores $top-k$ mais eficazes foram 5 (11 ocorrências), 15, 20, 25 e 30 (8 ocorrências cada) e 10 (7 ocorrências). Já para a métrica $CNPMI$, os melhores valores $top-k$ foram 15 (11 ocorrências), 20, 25, 30, 5 (9 ocorrências cada) e 10 (5 ocorrências), respectivamente.

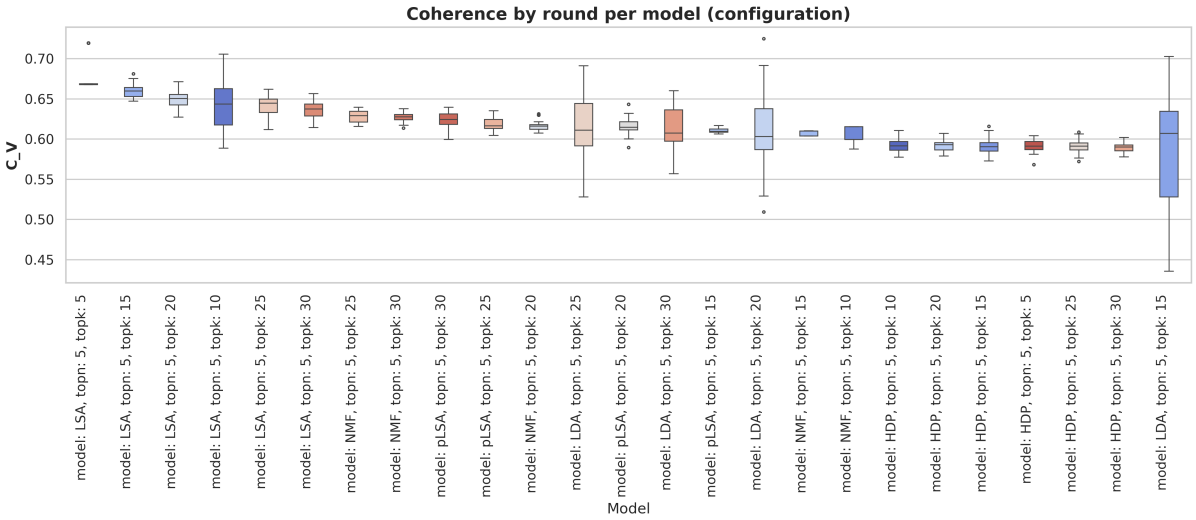


Figura 18 – *Boxplot* dos 25 modelos com maior coerência média CV.

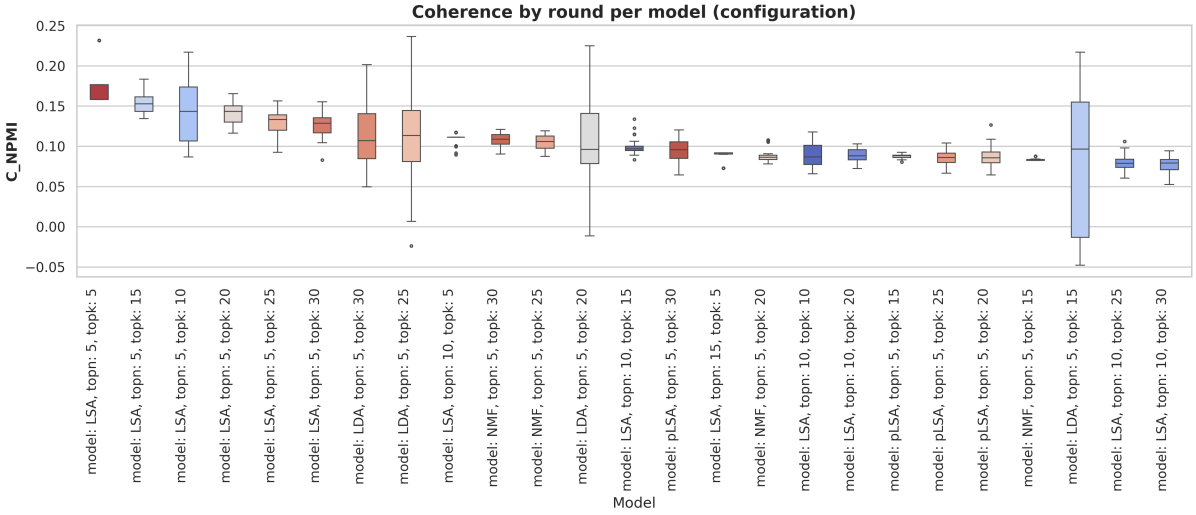


Figura 19 – *Boxplot* dos 25 modelos com maior coerência média $CNPMI$.

9.5.2 Avaliação Estatística

Para comparar o desempenho relativo dos algoritmos, são necessárias evidências estatísticas conclusivas. Assim, foram aplicados os testes de Anderson-Darling (AD) e Kolmogorov-Smirnov (KS). Os resultados indicam que certos modelos e configurações apresentam uma distribuição normal, conforme apresentado na Tabela 49 e na Tabela 50.

Tabela 49 – Resultados dos testes de normalidade (Anderson-Darling e Kolmogorov-Smirnov) para a métrica *CV* entre modelos e configurações. Valor crítico de 0,719 e N = 35. Mostrando apenas os 25 modelos com a maior coerência média, mas aplicado a todos os 216 modelos.

Config	Métrica	AD_Statistic	AD_Critical_5%	AD_Reject_at_5%	KS_Statistic	KS_pvalue
model: HDP, top-n: 5, top-k: 10	*CV*	0.284013	0.719	False	0.085657	9.401347e-01
model: HDP, top-n: 5, top-k: 15	*CV*	0.692053	0.719	False	0.138716	4.693841e-01
model: HDP, top-n: 5, top-k: 20	*CV*	0.300185	0.719	False	0.097755	8.594782e-01
model: HDP, top-n: 5, top-k: 25	*CV*	0.549617	0.719	False	0.139753	4.600110e-01
model: HDP, top-n: 5, top-k: 30	*CV*	0.202301	0.719	False	0.066194	9.952686e-01
model: HDP, top-n: 5, top-k: 5	*CV*	0.350576	0.719	False	0.083276	9.517420e-01
model: LDA, top-n: 5, top-k: 15	*CV*	0.623572	0.719	False	0.138019	4.757282e-01
model: LDA, top-n: 5, top-k: 20	*CV*	0.221110	0.719	False	0.081094	9.610818e-01
model: LDA, top-n: 5, top-k: 25	*CV*	0.263966	0.719	False	0.099633	8.441219e-01
model: LDA, top-n: 5, top-k: 30	*CV*	0.550680	0.719	False	0.106294	7.851221e-01
model: LSA, top-n: 5, top-k: 10	*CV*	0.894590	0.719	True	0.168011	2.473601e-01
model: LSA, top-n: 5, top-k: 15	*CV*	0.509020	0.719	False	0.107231	7.763647e-01
model: LSA, top-n: 5, top-k: 20	*CV*	0.324642	0.719	False	0.115306	6.978067e-01
model: LSA, top-n: 5, top-k: 25	*CV*	0.431313	0.719	False	0.128323	5.679148e-01
model: LSA, top-n: 5, top-k: 30	*CV*	0.410984	0.719	False	0.105012	7.969562e-01
model: LSA, top-n: 5, top-k: 5	*CV*	8.971564	0.719	True	0.476264	8.136301e-08
model: NMF, top-n: 5, top-k: 10	*CV*	5.425697	0.719	True	0.396011	1.840425e-05
model: NMF, top-n: 5, top-k: 15	*CV*	5.947324	0.719	True	0.412082	6.834794e-06
model: NMF, top-n: 5, top-k: 20	*CV*	2.130595	0.719	True	0.216880	6.331921e-02
model: NMF, top-n: 5, top-k: 25	*CV*	0.653413	0.719	False	0.121690	6.338602e-01
model: NMF, top-n: 5, top-k: 30	*CV*	0.195797	0.719	False	0.076759	9.760541e-01
model: pLSA, top-n: 5, top-k: 15	*CV*	1.900900	0.719	True	0.224083	5.026736e-02
model: pLSA, top-n: 5, top-k: 20	*CV*	0.611442	0.719	False	0.110636	7.437987e-01
model: pLSA, top-n: 5, top-k: 25	*CV*	0.478568	0.719	False	0.124746	6.033013e-01
model: pLSA, top-n: 5, top-k: 30	*CV*	0.291882	0.719	False	0.073135	9.851288e-01

Os resultados foram divididos em dois grupos: Grupo A, onde há evidências de que os resultados seguem uma distribuição normal, e Grupo B, onde não há evidências de normalidade. Para avaliar o desempenho dos modelos pareados no Grupo A, foi utilizado o teste t pareado, enquanto o teste de Wilcoxon com sinais ordenados foi aplicado para o Grupo B. A Tabela 51 descreve o número de vezes que o modelo avaliado superou os demais.

Os resultados apresentados na Tabela 51 demonstram de forma consistente o desempenho superior dos modelos baseados em Latent Semantic Analysis (LSA), especialmente nas configurações com menores valores de $top - n$ e $top - k$. Observa-se que as variações do LSA com $top - n = 5$ obtiveram as maiores pontuações em ambas as métricas de coerência — *CV* e $CNPMI$ —, alcançando os primeiros lugares no *ranking* geral. Em particular, a configuração com $top - n = 5$ e $top - k = 5$ destacou-se como a mais eficaz, superando todos os demais 215 métodos avaliados, o que evidencia a robustez e a estabilidade dessa parametrização.

A tendência observada indica que o desempenho do LSA diminui gradativamente à medida que o parâmetro $top - k$ aumenta, sugerindo que um número reduzido de termos mais

Tabela 50 – Resultados do teste de normalidade (Anderson–Darling e Kolmogorov–Smirnov) para a métrica $CNPMI$ em diferentes modelos e configurações. Valor crítico de 0,719 e N = 35.

Config	Metric	AD_Statistic	AD_Critical_5%	AD_Reject_at_5%	KS_Statistic	KS_pvalue
model: LDA, top-n: 5, top-k: 15	$CNPMI$	1.083289	0.719	True	0.150547	0.368681
model: LDA, top-n: 5, top-k: 20	$CNPMI$	0.324173	0.719	False	0.095154	0.879625
model: LDA, top-n: 5, top-k: 25	$CNPMI$	0.163557	0.719	False	0.067908	0.993511
model: LDA, top-n: 5, top-k: 30	$CNPMI$	0.539028	0.719	False	0.125273	0.598055
model: LSA, top-n: 10, top-k: 10	$CNPMI$	0.618090	0.719	False	0.128642	0.564785
model: LSA, top-n: 10, top-k: 15	$CNPMI$	2.326013	0.719	True	0.230684	0.040407
model: LSA, top-n: 10, top-k: 20	$CNPMI$	0.280068	0.719	False	0.101739	0.826180
model: LSA, top-n: 10, top-k: 25	$CNPMI$	0.214404	0.719	False	0.095555	0.876611
model: LSA, top-n: 10, top-k: 30	$CNPMI$	0.444250	0.719	False	0.117572	0.675169
model: LSA, top-n: 10, top-k: 5	$CNPMI$	5.658704	0.719	True	0.433285	0.000002
model: LSA, top-n: 15, top-k: 5	$CNPMI$	9.476502	0.719	True	0.438859	0.000001
model: LSA, top-n: 5, top-k: 10	$CNPMI$	0.970999	0.719	True	0.161610	0.287937
model: LSA, top-n: 5, top-k: 15	$CNPMI$	0.547248	0.719	False	0.127862	0.572444
model: LSA, top-n: 5, top-k: 20	$CNPMI$	0.461266	0.719	False	0.108281	0.766434
model: LSA, top-n: 5, top-k: 25	$CNPMI$	0.309672	0.719	False	0.100200	0.839363
model: LSA, top-n: 5, top-k: 30	$CNPMI$	0.310623	0.719	False	0.099127	0.848324
model: LSA, top-n: 5, top-k: 5	$CNPMI$	7.059189	0.719	True	0.405031	0.000011
model: NMF, top-n: 5, top-k: 15	$CNPMI$	4.997353	0.719	True	0.378835	0.000050
model: NMF, top-n: 5, top-k: 20	$CNPMI$	3.069506	0.719	True	0.275812	0.007611
model: NMF, top-n: 5, top-k: 25	$CNPMI$	0.576560	0.719	False	0.138278	0.473371
model: NMF, top-n: 5, top-k: 30	$CNPMI$	0.299215	0.719	False	0.113968	0.711104
model: pLSA, top-n: 5, top-k: 15	$CNPMI$	0.935512	0.719	True	0.200904	0.102791
model: pLSA, top-n: 5, top-k: 20	$CNPMI$	0.473631	0.719	False	0.099061	0.848867
model: pLSA, top-n: 5, top-k: 25	$CNPMI$	0.293222	0.719	False	0.085942	0.938645
model: pLSA, top-n: 5, top-k: 30	$CNPMI$	0.186404	0.719	False	0.083504	0.950693

representativos é suficiente para maximizar a coerência dos tópicos gerados. Esse comportamento é consistente com estudos prévios que apontam a sensibilidade do LSA à dimensionalidade e à seleção de termos relevantes.

Os métodos Non-negative Matrix Factorization (NMF) e Probabilistic Latent Semantic Analysis (pLSA) apresentaram desempenho intermediário, posicionando-se logo após as melhores configurações do LSA. Em especial, as configurações de NMF com $top-k$ entre 20 e 30 mostraram resultados competitivos, aproximando-se dos valores obtidos pelas variantes mais fortes do LSA. Já o Latent Dirichlet Allocation (LDA) apresentou resultados estáveis, mas ligeiramente inferiores aos de NMF e pLSA, enquanto o Hierarchical Dirichlet Process (HDP) obteve desempenho mais modesto, posicionando-se nas últimas colocações entre os 25 melhores modelos.

De modo geral, a análise estatística dos resultados confirma que as diferenças observadas entre os modelos não se devem ao acaso, sendo estatisticamente significativas tanto nos grupos de normalidade (testados pelo paired t-test) quanto nos grupos sem normalidade (avaliados pelo Wilcoxon Signed-Rank Test). Assim, conclui-se que o LSA, em especial nas configurações com $top-n = 5$ e $top-k \leq 15$, apresenta desempenho superior de forma consistente e estatisticamente robusta, configurando-se como o método mais eficaz para a geração de tópicos coerentes nas condições avaliadas.

Tabela 51 – Pontuações totais dos 25 melhores modelos por métrica de coerência (C_NPMIeC_V)$esomageral$.

Modelo	**C_NPMI**	**C_V**	**Total**
model: LSA, top-n: 5, top-k: 5	215	215	430
model: LSA, top-n: 5, top-k: 15	213	214	427
model: LSA, top-n: 5, top-k: 20	212	212	424
model: LSA, top-n: 5, top-k: 10	212	210	422
model: LSA, top-n: 5, top-k: 25	210	210	420
model: LSA, top-n: 5, top-k: 30	209	210	419
model: NMF, top-n: 5, top-k: 25	203	208	411
model: NMF, top-n: 5, top-k: 30	203	208	411
model: pLSA, top-n: 5, top-k: 30	201	206	407
model: LDA, top-n: 5, top-k: 30	204	199	403
model: LDA, top-n: 5, top-k: 25	203	199	402
model: NMF, top-n: 5, top-k: 20	194	201	395
model: pLSA, top-n: 5, top-k: 25	194	201	395
model: pLSA, top-n: 5, top-k: 15	194	200	394
model: pLSA, top-n: 5, top-k: 20	193	201	394
model: LDA, top-n: 5, top-k: 20	196	197	393
model: NMF, top-n: 5, top-k: 15	193	199	392
model: LSA, top-n: 10, top-k: 5	204	186	390
model: LSA, top-n: 10, top-k: 15	202	186	388
model: LSA, top-n: 10, top-k: 20	194	180	374
model: LSA, top-n: 10, top-k: 10	194	174	368
model: LSA, top-n: 15, top-k: 5	196	169	365
model: LSA, top-n: 10, top-k: 25	189	175	364
model: HDP, top-n: 5, top-k: 10	171	190	361
model: LSA, top-n: 10, top-k: 30	186	175	361

9.6 Considerações Finais

O processo de auditoria é tipicamente caracterizado como dispendioso, demorado e exigente em termos de recursos humanos e materiais. Nesse contexto, torna-se imperativo implementar soluções e técnicas que facilitem a automatização da análise de denúncias de corrupção. Esse processo é geralmente dividido em duas etapas: na primeira, o objetivo é identificar elementos e evidências de corrupção, como fornecedores, contratos, funcionários, clientes e outras partes interessadas, avaliando a plausibilidade e a consistência das denúncias e indícios de fraude; na segunda etapa, a investigação propriamente dita é conduzida.

Para construir o conhecimento necessário para as atividades de auditoria, é crucial coletar informações diretamente relacionadas aos objetivos da auditoria. Nessa etapa, diversas fontes são consultadas, incluindo páginas da web. Para aprimorar o processo de coleta de informações, técnicas de web scraping podem ser empregadas para extrair dados em larga escala de sites relacionados à saúde. Além disso, para apoiar a análise desse extenso volume de dados, técnicas de PNL, como sumarização de texto, podem ser utilizadas, reduzindo substancialmente o tempo e os recursos necessários para analisar e compilar evidências de potenciais irregularidades.

Neste contexto, com o objetivo de apoiar, aprimorar e otimizar a coleta de informações

relevantes que possam auxiliar no combate a irregularidades, este trabalho apresenta os resultados da aplicação de 216 modelos de modelagem de tópicos em um conjunto de notícias relacionadas à área da saúde com indícios de irregularidade. Buscou-se, assim, avaliar se tais métodos podem contribuir para o processo de auditoria, identificando diretamente quais são os temas (tópicos) das notícias que possuem indícios de irregularidade, bem como identificar qual é o melhor conjunto de parâmetros (configuração) e sua consistência em diversas execuções.

Neste experimento controlado **in vitro**, utilizando uma base de dados curada com 421 amostras de notícias, foram executados 216 modelos de modelagem de tópicos, repetidos em 35 rodadas para mensurar a coerência interna dos tópicos dos modelos de forma robusta por meio das métricas de coherence CV e $CNPMI$. Para responder as research questions, analisou-se quais são os melhores modelos comparando a coherence médio e o desempenho par a par entre os modelos. Para mensurar a consistência da coherence, verificou-se o desvio padrão dos resultados gerados pelos 216 modelos.

Entre os cinquenta modelos com os maiores valores médios de CV, o método LSA destacou-se, aparecendo 20 vezes, enquanto cada um dos outros métodos apareceu seis vezes. Da mesma forma, entre os cinquenta modelos com os maiores valores médios de $CNPMI$, o LSA foi novamente o mais frequente, com 19 ocorrências, seguido por NMF e pLSA com nove cada, LDA com sete e HDP com seis.

Os resultados apresentados na Tabela 51 demonstram consistentemente o desempenho superior dos modelos baseados em Análise Semântica Latente (LSA), particularmente em configurações com valores de $top-n$ e $top-k$ mais baixos. Observa-se que as variantes de LSA com $top-n = 5$ alcançaram as maiores pontuações em ambas as métricas de coerência, CV e $CNPMI$, garantindo as primeiras posições no *ranking* geral. Em particular, a configuração com ‘top-n = 5‘ e ‘top-k = 5‘ emergiu como a mais eficaz, superando todos os outros 215 métodos avaliados, evidenciando assim a robustez e a estabilidade dessa parametrização.

A tendência observada indica que o desempenho da Análise Semântica Latente (LSA) diminui gradualmente à medida que o parâmetro ‘top-k‘ aumenta, sugerindo que um conjunto menor de termos representativos é suficiente para maximizar a coerência temática. Esse comportamento está alinhado com estudos anteriores que destacam a sensibilidade da LSA à dimensionalidade e à seleção de termos relevantes.

Os métodos de Fatoração de Matrizes Não Negativas (NMF) e Análise Semântica Latente Probabilística (pLSA) apresentaram desempenho intermediário, classificando-se logo abaixo das melhores configurações de LSA. Notavelmente, as configurações de NMF com valores de ‘top-k‘ entre 20 e 30 produziram resultados competitivos, aproximando-se das pontuações alcançadas pelas variantes mais robustas da LSA. Em contraste, a Alocação Latente de Dirichlet (LDA) apresentou resultados estáveis, porém ligeiramente inferiores aos da NMF e da pLSA, enquanto o Processo Hierárquico de Dirichlet (HDP) demonstrou um desempenho mais modesto, figurando entre os piores dos 25 melhores modelos.

De modo geral, a análise estatística confirma que as diferenças observadas entre os modelos não se devem à variação aleatória, sendo estatisticamente significativas tanto para os grupos com distribuição normal (testados pelo teste t pareado) quanto para os grupos com distribuição não normal (avaliados pelo teste de Wilcoxon Signed-Rank Test). Portanto, pode-se concluir que a LSA, particularmente em configurações com $top - n = 5$ e $top - k \leq 15$, supera consistentemente e estatisticamente os demais métodos, consolidando-se como a abordagem mais eficaz para a geração de tópicos coerentes nas condições avaliadas.

Do ponto de vista do auditor, recomenda-se empregar os modelos que demonstraram maior coerência e consistência — ou seja, o LSA configurado com valores de $top - n$ e $top - k$ iguais a 5. Neste estudo, os métodos foram avaliados quantitativamente, sem a necessidade de intervenção humana na seleção de parâmetros.

A seguir, uma discussão sobre os capítulos do mapeamento sistemático da literatura, os métodos extrativos propostos e as avaliações experimentais realizadas é apresentada no Capítulo 10.

10 Discussão

Este capítulo concentra-se na conexão entre os achados do Mapeamento Sistemático da Literatura, métodos de resumo extrativo propostos e avaliações experimentais apresentadas. A discussão não apenas limita-se em justificar as decisões da pesquisa, mas aprofunda-se nas soluções apresentadas. O objetivo deste capítulo é demonstrar como as pesquisas apresentadas representam uma resposta sólida e tangível às lacunas da literatura e, também, expandem a base de conhecimento experimental da literatura, que permanece limitada no campo da Ciência da Computação, mas é bem estabelecida em outros campos.

10.1 Mapeamento Sistemático sobre Sumarização de Textos

O Mapeamento Sistemático da Literatura, apresentado no Capítulo 3, serviu como referência para orientar as etapas subsequentes da pesquisa. A análise aprofundada dos estudos selecionados permitiu identificar os principais métodos empregados, as áreas predominantes do domínio da saúde, os benefícios alcançados e os principais desafios reportados na literatura.

O objetivo do mapeamento consistiu em investigar os estudos mais recentes sobre sumarização de textos no domínio da saúde, por meio da análise dos métodos desenvolvidos, dos desafios enfrentados, dos benefícios obtidos e das áreas de aplicação.

A análise dos artigos selecionados no mapeamento evidenciou a predominância de métodos tradicionais baseados em *Computational Linguistics*, bem como o uso majoritário de abordagens híbridas, que combinam técnicas de sumarização de texto com um ou mais métodos auxiliares, tais como classificação de texto, modelagem de tópicos e reconhecimento de entidades nomeadas (NER), entre outros. Além disso, observou-se que os trabalhos diferem significativamente entre si, empregando técnicas distintas para a geração dos resumos. No que se refere às áreas da saúde, constatou-se uma ampla diversidade de domínios de aplicação, o que demonstra a flexibilidade e a aplicabilidade dos métodos em diferentes contextos. Esses estudos

evidenciam uma demanda crescente por soluções voltadas à otimização e melhoria da busca por informações, ao apoio à tomada de decisão, à melhoria de processos operacionais, bem como à redução do tempo de análise e da sobrecarga informacional, entre outros benefícios identificados.

A maioria dos trabalhos aborda o problema de forma orientada à consultas e considerando documentos únicos, o que evidencia a limitação de escalabilidade para o processamento de grandes volumes de texto. Ademais, a maior parte dos estudos trata o problema por meio de abordagens extrativas, indicando uma lacuna de pesquisa no uso de métodos abstrativos.

Somadas a essas limitações e lacunas, o mapeamento também revelou desafios recorrentes no desenvolvimento e/ou na implementação de métodos de resumo automático de texto, tais como: custos computacionais e de processamento, qualidade dos dados, escassez e limitações de bases de dados, dependência de especialistas, limitações inerentes aos métodos propostos, restrições das métricas de avaliação e a complexidade linguística dos textos analisados.

Com base nesses achados, buscou-se preencher as lacunas identificadas na literatura e mitigar as limitações observadas, abordando ambas as formas de sumarização: extrativa e abstrativa. No âmbito da abordagem extrativa, objetivou-se mitigar as limitações dos métodos existentes por meio da proposta, avaliação e implementação de novos métodos que não apresentem as mesmas restrições. Já no contexto da abordagem abstrativa, buscou-se suprir a lacuna de conhecimento identificada na literatura por meio da avaliação experimental do uso de *Small Language Models*. Adicionalmente, considerando o amplo emprego de abordagens híbridas, também foram avaliados métodos auxiliares de classificação de texto e modelagem de tópicos, possibilitando tanto o uso isolado da sumarização no contexto de auditoria quanto sua integração com técnicas de classificação e/ou modelagem de tópicos.

10.2 Resumo Extrativo de Texto

Abordagens baseadas em *Machine Learning* ou *Deep Learning*, embora mais recentes quando comparadas às técnicas de linguística computacional, demandam o retreinamento dos modelos sempre que é atingido um determinado limiar de erro aceitável. Considerando o contexto do problema, o domínio da saúde, reconhecidamente mais complexo em virtude de sua linguagem especializada (ADAMS et al., 2021), tal processo exigiria a anotação das bases de dados por especialistas da área. Ademais, ao se utilizar notícias como fonte de dados, observa-se que o foco temático e a forma de descrição dos eventos sofrem variações frequentes. Por exemplo, ocorrências como denúncias ou suspeitas de corrupção em determinadas localidades tendem a configurar eventos pontuais em um dado intervalo de tempo, os quais não seriam capturados de maneira dinâmica e recorrente por processos de anotação estática das bases de dados.

A tarefa de resumo de texto extrativo é, essencialmente, um problema de otimização combinatória. Nesta tarefa, tem-se que selecionar um subconjunto de frases que melhor represente o texto. Diante disso e partindo da lacuna sobre métodos extrativos, propõem-se o MRMRSFLA

e o HSSFLA, dois métodos baseados em otimização combinatória e otimizados por meio de *swarm intelligence*.

As limitações identificadas no mapeamento e na revisão da literatura foram abordadas pelos métodos MRMRSFLA e HSSFLA, uma vez que ambos foram avaliados de forma robusta por meio da definição de seus parâmetros com o conjunto de dados DUC2001 e da mensuração de desempenho no DUC2002, evidenciando sua capacidade de generalização. No que se refere à complexidade, os métodos constituem extensões do *Shuffled Frog-Leaping Algorithm*, um algoritmo memético caracterizado por baixa complexidade computacional e, consequentemente, por reduzida demanda de recursos computacionais. Além disso, ambos os métodos são não supervisionados, não dependem de especialistas em nenhuma etapa do *pipeline*, foram concebidos como abordagens genéricas e são aplicáveis tanto a tarefas de sumarização de documento único quanto de múltiplos documentos, independentemente do domínio ou do idioma.

Para ambos os métodos, foram conduzidos experimentos usando os conjuntos de dados de benchmark DUC2001 e DUC2002. E por meio de análises estatísticas, temos evidências de que proporcionam resultados superiores em comparação com outras abordagens na literatura.

10.3 Avaliação Experimental

Partindo da lacuna de pesquisas sobre o uso de métodos abstrativos, conduzimos uma avaliação experimental no contexto da saúde focada no contexto de auditoria, buscando auxiliar diretamente a etapa de levantamento de informações sobre o objetivo da auditoria por meio do uso de *Small Language Models* para minimizar a sobrecarga informacional.

Este experimento teve como objetivo avaliar métodos automáticos de sumarização de textos por meio da comparação da qualidade de resumos gerados por máquinas com aqueles produzidos por humanos, sob a perspectiva de Cientistas de Dados e Auditores do SUS, no contexto de auditorias realizadas pelo Departamento Nacional de Auditoria do Sistema Único de Saúde (Sistema Único de Saúde — SUS) (AudSUS).

Entre os métodos com melhor desempenho relacionado à qualidade dos resumos e relativos aos demais, destacam-se os modelos **NousResearch/Hermes-3-Llama-3.2-3B**, **Qwen/Qwen2.5-7B-Instruct** e **meta-llama/Llama-3.2-3B-Instruct**. Esses métodos se sobressaem de forma consistente nas diferentes métricas de avaliação, demonstrando capacidade superior em capturar e preservar o significado contextual do texto original, além de sintetizar adequadamente as principais informações, quando comparados ao desempenho humano.

De forma complementar, e seguindo a tendência de métodos híbridos identificada no mapeamento, também avaliou-se experimentalmente métodos de classificação e modelagem de tópicos. Todos os experimentos foram avaliados estatisticamente por suas respectivas métricas, reforçando o rigor das pesquisas.

10.4 Perspectiva do Auditor

Fraude e corrupção figuram entre os principais crimes que afetam as instituições públicas, sendo o setor da saúde particularmente vulnerável em virtude de sua elevada complexidade estrutural, da coexistência de provedores públicos e privados, do grande número de atores envolvidos, da natureza globalizada das cadeias de suprimentos, dos elevados custos financeiros e da acentuada assimetria de informações entre as partes interessadas. Esses fatores contribuem para a fragilização dos sistemas de saúde, resultando em desperdício de recursos, redução da capacidade de resposta e da resiliência frente a emergências médicas, bem como na restrição do acesso a serviços essenciais.

Sob a perspectiva de um auditor, a utilização de métodos de resumo de texto extrativos ou abstrativos, híbridos ou individualizados, não é apenas recomendável mas factível conforme as evidências estatísticas apresentadas. Os métodos de resumo, extrativos e abstrativos, demonstram ser tecnicamente confiáveis, uma vez que apresentam consistência nos resultados, destacam as informações mais relevantes e mantêm um tamanho médio de resumo comparável ao desempenho humano. Assim, por meio de métodos automatizados e ao contribuir para a redução da sobrecarga informacional, esses modelos podem apoiar o processo de auditoria na fase analítica, aumentando a eficácia e a efetividade no levantamento de informações e possibilitando a preparação das equipes para a fase operativa em menor tempo.

Quanto a escolha entre abordagem extrativa e abstrativa, é necessário avaliar a disponibilidade de recursos e urgência no levantamento de informações. Para cenários de maior urgência, recomenda-se o uso de métodos extrativos (HSSFLA ou MRMRSFLA), enquanto em cenários com maior tempo disponível para levantamento e análise das informações, recomenda-se o uso dos métodos abstrativos, a exemplo dos Small Language Models (SLMs).

Em relação a escolha dos métodos para a classificação automatizada de notícias com indícios de irregularidade, depende de fatores como factualidade, urgência, disponibilidade de recursos computacionais, tempo de inferência e o tipo de aplicação (processamento em fluxo contínuo ou em lote). Em cenários que demandam maior precisão na identificação de notícias com indícios de irregularidades, conforme avaliado por métricas como *accuracy*, *precision*, *recall* e *F1-score*, é preferível empregar modelos que apresentem desempenho superior nesses indicadores, como *Random Forest* (validação cruzada padrão), *Random Forest* (validação cruzada aleatória) e *SVC* (validação cruzada aleatória), os quais se destacaram em todas as métricas de avaliação do experimento. Em situações que exigem maior urgência e agilidade no planejamento, recomenda-se a utilização de modelos com menor complexidade teórica em termos de treinamento, predição e uso de memória, tais como *Naive Bayes*, *Logistic Regression* ou *LightGBM*.

Já nos métodos de modelagem de tópicos, para identificar os temas abordados nas notícias, e, assim resumi-las em lotes, recomenda-se o uso dos melhores modelos quanto à coherence e sua consistência, ou seja, LSA utilizando os parâmetros top-n e top-k com o valor 5. Essa abordagem

aprimora a eficácia geral das auditorias e facilita uma preparação mais ágil das equipes para a etapa operacional.

11
Conclusões

Este trabalho de dissertação partiu do desafio no processo de auditoria, que, além de sua complexidade inerente, enfrenta o problema de sobrecarga informacional na etapa de levantamento de evidências de corrupção devido ao alto volume de dados gerados diariamente. A metodologia deste trabalho foi estruturada em três etapas principais. A primeira consistiu em um mapeamento sistemático da literatura (MSL) com objetivo identificar o estado da arte das pesquisas sobre métodos de resumo automático de texto aplicados no contexto da saúde, seus desafios enfrentados, dos benefícios alcançados e das áreas de aplicação. Visando alcançar o objetivo principal da pesquisa, a segunda etapa consistiu na proposição de métodos de resumo de texto extrativos. A terceira etapa, consistiu na execução de um experimento controlado in vitro, avaliando métodos abstrativos e auxiliares.

A análise dos trabalhos por meio de um mapeamento sistemático e revelou a predominância de métodos de resumo baseados em linguística computacional, métodos híbridos e extrativos. No entanto, o estudo também expôs limitações como o foco em resumos baseados em query e de documentos únicos (evidenciando limitações sobre escalabilidade), altos custos computacionais e de processamento, baixa qualidade dos dados, escassez e limitações de bases de dados, dependência de especialistas, limitações inerentes aos métodos propostos, restrições das métricas de avaliação e a complexidade linguística dos textos analisados.

Endereçando essas lacunas e limitações por meio de métodos extrativos, na segunda etapa os métodos MRMRSFLA e HSSFLA foram propostos, implementados e avaliados, trazendo robustez na avaliação, generalização de resultados, baixa complexidade e consumo computacional, são não supervisionados, genéricos e multi-documentos. E por meio de análises estatísticas, obteve-se evidências de que proporcionam resultados superiores em comparação com outras abordagens na literatura.

Já na terceira etapa, abordando o problema por meio de métodos abstrativos, conduzimos uma avaliação experimental no contexto da saúde focada no contexto de auditoria, buscando

auxiliar diretamente a etapa de levantamento de informações sobre o objetivo da auditoria por meio do uso de *Small Language Models* para minimizar a sobrecarga informacional. De forma complementar, e seguindo a tendência de métodos híbridos identificada no mapeamento, também avaliou-se experimentalmente métodos de classificação e modelagem de tópicos. Todos os experimentos foram avaliados estatísticamente por suas respectivas métricas, reforçando o rigor das pesquisas.

11.0.1 Contribuições

As principais contribuições desta dissertação consistem na demonstração, por meio de análises estatísticas e avaliações experimentais, da viabilidade de adoção de métodos de resumo extrativo (MRMRSFLA e HSSFLA) e abstrativo (SMLs) para a mitigação do problema de sobrecarga informacional no contexto de auditorias em saúde. Cada etapa do estudo foi estruturada com rigor metodológico e consistência estatística, possibilitando a replicabilidade dos experimentos a partir da base de dados pública disponibilizada e do código-fonte aberto. Adicionalmente, destacam-se as seguintes contribuições, materializadas em seis produções científicas:

1. **Mapeamento Sistemático da Literatura**: mapeamento sistemático do estado da arte sobre métodos de resumo de texto no contexto da saúde, investigando os estudos mais recentes, analisando os métodos desenvolvidos, os desafios enfrentados, os benefícios alcançados e as áreas de aplicação. Este artigo, *A Systematic Mapping of Text Summarization Methods Applied in Health Domain*, foi submetido para o XXII Simpósio Brasileiro de Sistemas de Informação (SBSI).

2. **Métodos de resumo extrativo:** proposição e avaliação de novos métodos de resumo extrativos MRMRSFLA e HSSFLA:.

    2.1. O MRMRSFLA foi introduzido no artigo *A Topic Based Generic Extractive Multi-document Text Summarization Method Using Memetic Algorithm and Combinatorial Optimization*, submetido ao periódico *Memetic Computing*;

    2.2. Enquanto que o HSSFLA foi apresentado no artigo *A Generic Extractive Multi-document Text Summarization Method Using Memetic Algorithm and Combinatorial Optimization*, **aceito** pelo XXII Simpósio Brasileiro de Sistemas de Informação (SBSI).

3. **Avaliação Experimental:** os 3 artigos que avaliam experimentalmente SLMs, métodos de classificação e modelagem de tópicos expandem a base de conhecimento experimental:

    3.1. *Small Language Models Applied in Text Summarization Task of Health-Related News to Improve Public Health Audit: An Experimental Case Study*, **publicado** no periódico *Frontiers in Artificial Intelligence*;

3.2. *Experimental Evaluation of Machine Learning Algorithms for Classifying Health-Related News with Indications of Irregularity*, **aceito** no XXII Simpósio Brasileiro de Sistemas de Informação (SBSI);

3.3. *Experimental Evaluation of Topic Modeling Methods for Categorizing Irregularities in Health-related news*, **aceito** na conferência 7th International Conference on Computational Processing of Portuguese (PROPOR 2026).

11.0.2 Recomendações

Esta dissertação teve como objetivo desenvolver e avaliar Métodos de Resumo Automático de Texto, voltado ao apoio de auditorias no setor de saúde. Com este objetivo e diante dos resultados levantados nesta dissertação, recomenda-se:

1. Em cenários com poucos recursos financeiros e/ou computacionais e que a factualidade dos resultados deve ser priorizada, recomenda-se utilizar os métodos de resumos extrativos (HSSFLA ou MRMRSFLA). Caso tais recursos estejam disponíveis, recomenda-se a utilização métodos abstrativos, a exemplo dos SMLs avaliados experimentalmente neste trabalho;

2. Implementar um ou mais métodos auxiliares para classificação automática de notícias com indícios de irregularidade e/ou aplicar modelagem de tópicos para agrupar as notícias tematicamente, como uma etapa anterior a de sumarização, potencializando seus resultados ao automatizar a anotação reduzindo o volume de informações iniciais e/ou agrupando tematicamente os resumos.

11.0.3 Limitações

Apesar dos resultados promissores obtidos, este estudo apresenta limitações que devem ser reconhecidas. Como:

1. **Representação de sentenças**: a representação de sentenças utilizadas no desenvolvimento dos métodos MRMRSFLA e HSSFLA baseia-se no esquema TF-IDF, que, embora eficaz, não possui a capacidade de capturar relações semânticas proporcionada por métodos contemporâneos baseados em *embeddings*, incluindo arquiteturas Transformer. Essa limitação pode reduzir a expressividade e a fidelidade contextual das representações vetoriais usadas em cálculos de similaridade.

2. **Avaliação**: a avaliação do MRMRSFLA e HSSFLA baseou-se exclusivamente em métricas ROUGE orientadas à *Recall*, sem incorporar medidas complementares que abordassem precisão, fluência ou adequação semântica. Essas limitações destacam oportunidades para pesquisas futuras visando aprimorar a robustez, a generalização e a validade comparativa do modelo por meio do uso de métricas contextuais, como o BERTscore.

11.0.4 Trabalhos Futuros

Quanto a trabalhos futuros, pretende-se investigar:

1. **Avaliação Experimental In Vivo**: Após validação por meio de experimentos em cada uma das etapas realizada neste trabalho, pode-se avaliar o desempenho dos métodos de resumo extrativo e/ou abstrativos combinados com métodos de classificação e/ou modelagem de tópicos num ambiente *in vivo*, ou seja, em produção.

2. **Melhorias nos Métodos Extrativos**: pretende-se aprofundar a investigação do problema de otimização por meio da exploração de outros métodos de representação de sentenças, como abordagens baseadas em *embeddings*, bem como ampliar o **benchmarking** por meio da comparação com outros tipos de algoritmos, tais como *deep learning*, *machine learning* e abordagens de sumarização baseadas em linguística computacional. Ademais, planeja-se otimizar a função objetivo por meio do uso de novas estratégias de mutação e da aplicação de diferentes algoritmos de inteligência de enxame, incorporando estratégias de mutação diversas e comparando os respectivos resultados.

Referências

ABBASI-GHALEHTAKI, R.; KHOTANLOU, H.; ESMAEILPOUR, M. Fuzzy evolutionary cellular learning automata model for text summarization. *Swarm and Evolutionary Computation*, Elsevier B.V., v. 30, p. 11 – 26, 2016. ISSN 22106502. Citado 4 vezes nas páginas 71, 72, 73 e 74.

ABULKHAIR, M. et al. Intelligent integration of discharge summary: A formative model. In: *2013 4th International Conference on Intelligent Systems, Modelling and Simulation*. [S.l.: s.n.], 2013. p. 99–104. ISSN 2166-0670. Citado 7 vezes nas páginas 59, 63, 64, 65, 67, 68 e 69.

ADAMS, G. et al. What's in a summary? laying the groundwork for advances in hospital-course summarization. In: TOUTANOVA, K. et al. (Ed.). *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Online: Association for Computational Linguistics, 2021. p. 4794–4811. Citado 8 vezes nas páginas 59, 64, 65, 66, 67, 68, 69 e 161.

AFZAL, M. et al. Clinical context–aware biomedical text summarization using deep neural network: Model development and validation. *Journal of Medical Internet Research*, JMIR Publications Inc., v. 22, n. 10, 2020. ISSN 14388871. Citado 7 vezes nas páginas 59, 63, 64, 65, 67, 68 e 69.

AL-RADAIDEH, Q. A.; BATAINEH, D. Q. A hybrid approach for arabic text summarization using domain knowledge and genetic algorithms. *Cognitive Computation*, Springer New York LLC, v. 10, n. 4, p. 651 – 669, 2018. ISSN 18669956. Citado 4 vezes nas páginas 71, 72, 73 e 74.

ALGULIEV, R. M.; ALIGULIYEV, R. M.; ISAZADE, N. R. Desamc+docsum: Differential evolution with self-adaptive mutation and crossover parameters for multi-document summarization. *Knowledge-Based Systems*, v. 36, p. 21 – 38, 2012. ISSN 09507051. Citado 3 vezes nas páginas 71, 72 e 73.

ALGULIEV, R. M.; ALIGULIYEV, R. M.; MEHDIYEV, C. A. Sentence selection for generic document summarization using an adaptive differential evolution algorithm. *Swarm and Evolutionary Computation*, v. 1, n. 4, p. 213 – 222, 2011. ISSN 22106502. Citado 2 vezes nas páginas 71 e 72.

ALGULIYEV, R. M.; ALIGULIYEV, R. M.; ISAZADE, N. R. An unsupervised approach to generating generic summaries of documents. *Applied Soft Computing Journal*, Elsevier Ltd, v. 34, p. 236 – 250, 2015. ISSN 15684946. Citado 8 vezes nas páginas 33, 34, 71, 72, 79, 81, 98 e 100.

ALIGULIYEV, R. M. A new sentence similarity measure and sentence based extractive technique for automatic text summarization. *Expert Systems with Applications*, v. 36, n. 4, p. 7764 – 7772, 2009. ISSN 09574174. Citado 4 vezes nas páginas 71, 72, 73 e 74.

ALQAISI, R.; GHANEM, W.; QAROUSH, A. Extractive multi-document arabic text summarization using evolutionary multi-objective optimization with k-medoid clustering. *IEEE Access*, Institute of Electrical and Electronics Engineers Inc., v. 8, p. 228206 – 228224, 2020. ISSN 21693536. Citado 6 vezes nas páginas 71, 72, 73, 74, 93 e 110.

AMARAL, J.; RODRIGUES, J. Alocação de tópicos latentes — um modelo para segmentação de dados de auditoria do governo de pe. *Revista de Engenharia e Pesquisa Aplicada*, v. 5, n. 1, p. 40–49, 2020. Citado 4 vezes nas páginas 36, 37, 38 e 39.

AMARAL, J. A. A. do et al. Alocacao de topicos latentes — um modelo para segmentacao de dados de auditoria do governo de pe. *null*, 2020. Citado na página 26.

ANGELOV, D. *Top2Vec: Distributed Representations of Topics*. 2020. Citado 2 vezes nas páginas 38 e 39.

ARSLAN, M.; CRUZ, C. Extracting business insights through dynamic topic modeling and ner. In: D., A.; J., D.; J., F. (Ed.). [S.l.]: Science and Technology Publications, Lda, 2022. v. 2, p. 215 – 222. ISBN 978-989758614-9. ISSN 21843228. Citado na página 33.

ASH, E.; GALLETTA, S.; GIOMMONI, T. *A Machine Learning Approach to Analyzing Corruption in Local Public Finances*. [S.l.], 2020. (Center for Law & Economics Working Paper Series, 06/2020). Open Access. In Copyright - Non-Commercial Use Permitted. Citado 2 vezes nas páginas 36 e 37.

BALAN, P. F.; GERITS, A.; VANDUFFEL, W. A practical application of text mining to literature on cognitive rehabilitation and enhancement through neurostimulation. *Front. Syst. Neurosci.*, Frontiers Media SA, v. 8, p. 182, set. 2014. Citado 7 vezes nas páginas 59, 62, 64, 65, 67, 68 e 69.

BANNUR, C. et al. Paacda: Comprehensive data corruption detection algorithm. *IEEE Access*, v. 11, p. 24908–24934, 2023. Citado na página 38.

BASILI, V. R.; WEISS, D. M. A methodology for collecting valid software engineering data. *IEEE Transactions on Software Engineering*, SE-10, n. 6, p. 728–738, 1984. Citado 3 vezes nas páginas 113, 132 e 146.

BENJELLOUN, F.-Z. et al. An overview of big data opportunities, applications and tools. *null*, 2015. Citado 2 vezes nas páginas 24 e 26.

BERAWI, M. et al. Digital innovation: Creating competitive advantages. *International Journal of Technology*, 2020. Citado na página 23.

BLAGOEVA, K. T.; BELSOSKA, M. M. Developing data driven products in the emerging markets. *KNOWLEDGE INTERNATIONAL JOURNAL*, 2019. Citado na página 23.

BLEI, D.; LAFFERTY, J. et al. Correlated topic models. *Advances in neural information processing systems*, MIT; 1998, v. 18, p. 147, 2006. Citado na página 39.

BLEI, D. M.; NG, A. Y.; JORDAN, M. I. Latent dirichlet allocation. *J. Mach. Learn. Res.*, JMLR.org, v. 3, n. null, p. 993–1022, mar. 2003. ISSN 1532-4435. Citado 3 vezes nas páginas 39, 40 e 41.

CABELLO-COLLADO, C. et al. Automated generation of clinical reports using sensing technologies with deep learning techniques. *Sensors (Basel)*, v. 24, n. 9, abr. 2024. Citado 7 vezes nas páginas 59, 62, 64, 65, 67, 68 e 69.

CAI, J. et al. RegEMR: a natural language processing system to automatically identify premature ovarian decline from chinese electronic medical records. *BMC Med. Inform. Decis. Mak.*, v. 23, n. 1, p. 126, jul. 2023. Citado 8 vezes nas páginas 59, 61, 64, 65, 66, 67, 68 e 69.

CAO, Y. et al. AskHERMES: An online question answering system for complex clinical questions. *J. Biomed. Inform.*, v. 44, n. 2, p. 277–288, abr. 2011. Citado 6 vezes nas páginas 59, 62, 64, 67, 68 e 69.

Capes. *Portal de periódicos CAPES/MEC [Journal Portal CAPES/MEC]*. 2024. Disponível em: https://www.periodicos.capes.gov.br. Acesso em: 12 de dezembro de 2024. Citado na página 54.

CARVALHO, M. *Gastos com saúde chegam a quase 9,7% do PIB, diz diretor do Ministério da Saúde*. 2024. Portal JOTA. Dados sobre a divisão de 60% para saúde suplementar e 40% para o SUS. Acesso em: 11 mar. 2026. Disponível em: <https://www.jota.info/saude/gastos-com-saude-chegam-a-quase-97-do-pib-diz-diretor-do-ministerio-da-saude>. Citado na página 25.

CASELI, H. M.; NUNES, M. G. V. (Ed.). *Processamento de Linguagem Natural: Conceitos, Técnicas e Aplicações em Português*. 3. ed. [S.l.]: BPLN, 2024. ISBN 978-65-01-20581-6. Citado na página 32.

CHAVES, A.; KESIKU, C.; GARCIA-ZAPIRAIN, B. Automatic text summarization of biomedical text data: A systematic review. *Information*, v. 13, n. 8, 2022. ISSN 2078-2489. Citado na página 61.

CHEN, Y. et al. Burextract-llama: An llm for clinical concept extraction in breast ultrasound reports. In: *Proceedings of the 1st International Workshop on Multimedia Computing for Health and Medicine*. New York, NY, USA: Association for Computing Machinery, 2024. (MCHM'24), p. 53–58. ISBN 9798400711954. Citado 6 vezes nas páginas 59, 62, 64, 67, 68 e 69.

Colaço JÚNIOR, M. *IA para a Galera Toda: Agentes e Inovação Experimental Sem Código*. [S.l.]: Amazon Publishing, 2025. ISBN 978-65-01-24603-1. Citado 7 vezes nas páginas 9, 47, 48, 113, 116, 132 e 146.

Colaço JÚNIOR, M. et al. Evaluation of a process for the experimental development of data mining, ai and data science applications aligned with the strategic planning. *Journal of Information Systems and Technology Management*, v. 19, Nov. 2022. Citado 2 vezes nas páginas 132 e 146.

Conselho Nacional de Secretários de Saúde. *Financiamento do Sistema Único de Saúde - Perspectivas dos estados e municípios*. Brasília, DF, 2024. Acesso em: 11 mar. 2026. Disponível em: <https://www.conass.org.br/>. Citado na página 25.

Controladoria-Geral da União, B. *Relatório de Auditoria Anual de Contas: Ministério da Saúde (Exercício 2023)*. Brasília, DF, 2024. Análise das demonstrações financeiras e conformidade do Ministério da Saúde. Acesso em: 11 mar. 2026. Disponível em: <https://eaud.cgu.gov.br/relatorios/download/1324796>. Citado na página 25.

CORRÊA, N. K. et al. *TeenyTinyLlama: open-source tiny language models trained in Brazilian Portuguese*. [S.l.]: Springer, 2024. Citado 2 vezes nas páginas 35 e 36.

CORRÊA, N. K. et al. *Tucano: Advancing Neural Text Generation for Portuguese*. 2024. Citado na página 115.

CORRêA, N. K. et al. TeenyTinyLlama: Open-source tiny language models trained in Brazilian Portuguese. *Machine Learning with Applications*, v. 16, p. 100558, 2024. ISSN 2666-8270. Citado na página 115.

CORRêA, N. K. et al. Tucano: Advancing neural text generation for portuguese. *Patterns*, p. 101325, 2025. ISSN 2666-3899. Citado na página 36.

DAMIANO, R. et al. Corruption detection through textual analysis: Evidence from eurozone banks. *Business Ethics, the Environment & Responsibility*, John Wiley & Sons Ltd, v. 0, p. 1–21, 2025. Open Access, Creative Commons Attribution License. Citado 2 vezes nas páginas 37 e 38.

DEERWESTER, S. et al. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, Wiley, v. 41, n. 6, p. 391–407, 1990. First published: September 1990, Citations: 6,569. Citado 2 vezes nas páginas 39 e 40.

EL-KASSAS, W. S. et al. Automatic text summarization: A comprehensive survey. *Expert Systems with Applications*, v. 165, p. 113679, 2021. ISSN 0957-4174. Citado 3 vezes nas páginas 43, 77 e 96.

ERKAN, G.; RADEV, D. R. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, 2004. Citado na página 115.

EUSUFF, M.; LANSEY, K.; PASHA, F. Shuffled frog-leaping algorithm: A memetic meta-heuristic for discrete optimization. *Engineering Optimization*, Taylor & Francis, v. 38, n. 2, p. 129–154, 2006. Published online: 25 Jan 2007, Received: 29 Sep 2004. Citado 2 vezes nas páginas 82 e 100.

FONTES, R. S. et al. Sussurro - detecção na web de eventos auditáveis que representam riscos à saúde pública. *Anais Estendidos do XXIII Simpósio Brasileiro de Computação Aplicada à Saúde (SBCAS 2023)*, 2023. Citado 7 vezes nas páginas 26, 47, 48, 49, 114, 133 e 147.

Gemma Team, T. M. et al. Gemma. Kaggle, 2024. Citado na página 115.

GROOTENDORST, M. Bertopic: Neural topic modeling with a class-based tf-idf procedure. *arXiv preprint arXiv:2203.05794*, 2022. Citado 4 vezes nas páginas 38, 39, 40 e 42.

GUIMARÃES, A. et al. *Health Related News Dataset*. Zenodo, 2025. Disponível em: <https://doi.org/10.5281/zenodo.18039964>. Citado na página 50.

GULDEN, C. et al. Extractive summarization of clinical trial descriptions. *Int. J. Med. Inform.*, Elsevier BV, v. 129, p. 114–121, set. 2019. Citado 8 vezes nas páginas 59, 63, 64, 65, 66, 67, 68 e 69.

HAGHIGHI, A.; VANDERWENDE, L. Exploring content models for multi-document summarization. In: *NAACL HLT 2009 - Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Proceedings of the Conference*. [S.l.: s.n.], 2009. Citado na página 115.

HAN, S.; HAO, X.; HUANG, H. An event-extraction approach for business analysis from online chinese news. *Electronic Commerce Research and Applications*, Elsevier B.V., v. 28, p. 244 – 260, 2018. ISSN 15674223. Citado na página 24.

HERNANDEZ-CASTANEDA, A. et al. Extractive automatic text summarization based on lexical-semantic keywords. *IEEE Access*, Institute of Electrical and Electronics Engineers Inc., v. 8, p. 49896 – 49907, 2020. ISSN 21693536. Citado 2 vezes nas páginas 71 e 72.

HO, H.-T. et al. Bio-inspired algorithms in nlp techniques: Challenges, limitations and its applications. *Computers, Materials and Continua*, v. 83, n. 3, p. 3945–3973, 2025. ISSN 1546-2218. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1546221825004461>. Citado na página 34.

HOFMANN, T. *Probabilistic Latent Semantic Analysis*. 2013. Citado 3 vezes nas páginas 39, 40 e 41.

HOSSAIN, E. et al. Natural language processing in electronic health records in relation to healthcare decision-making: A systematic review. *Computers in Biology and Medicine*, v. 155, p. 106649, 2023. ISSN 0010-4825. Citado na página 61.

HUANG, L. et al. Modeling document summarization as multi-objective optimization. In: . [S.l.: s.n.], 2010. p. 382 – 386. ISBN 978-076954020-7. Citado 4 vezes nas páginas 33, 71, 72 e 74.

JIANG, Y. et al. Natural language processing adoption in governments and future research directions: A systematic review. *Applied Sciences*, 2023. Citado 3 vezes nas páginas 24, 38 e 39.

JOUDAKI, H. et al. Using data mining to detect health care fraud and abuse: A review of literature. *Global Journal of Health Science*, v. 7, n. 1, p. 194–202, 2015. ISSN 1916-9736. Open Access under Creative Commons Attribution 4.0 License. Citado 2 vezes nas páginas 36 e 37.

KAR, A. K. Bio inspired computing – a review of algorithms and scope of applications. *Expert Systems with Applications*, v. 59, p. 20–32, 2016. ISSN 0957-4174. Disponível em: <https://www.sciencedirect.com/science/article/pii/S095741741630183X>. Citado na página 34.

KARYSTIANIS, G. et al. Automatic mining of symptom severity from psychiatric evaluation notes. *Int. J. Methods Psychiatr. Res.*, Wiley, v. 27, n. 1, p. e1602, mar. 2018. Citado 7 vezes nas páginas 59, 62, 64, 65, 67, 68 e 69.

KHURANA, A.; BHATNAGAR, V. Investigating entropy for extractive document summarization. *Expert Systems with Applications*, v. 187, p. 115820, 2022. ISSN 0957-4174. Citado 2 vezes nas páginas 80 e 99.

KIM, Y. et al. Competitive intelligence in social media twitter: Iphone 6 vs. galaxy s5. *Online Information Review*, Emerald Group Publishing Ltd., v. 40, n. 1, p. 42 – 61, 2016. ISSN 14684527. Citado na página 24.

KIRMANI, M. et al. Biomedical semantic text summarizer. *BMC Bioinformatics*, v. 25, n. 1, p. 152, abr. 2024. Citado 7 vezes nas páginas 59, 63, 64, 66, 67, 68 e 69.

KITCHENHAM, B. *Procedures for Performing Systematic Reviews*. Keele, Staffs, ST5 5BG, UK, 2004. © Kitchenham, 2004. Citado 2 vezes nas páginas 52 e 53.

KOSE, I.; GOKTURK, M.; KILIC, K. An interactive machine-learning-based electronic fraud and abuse detection system in healthcare insurance. *Applied Soft Computing*, v. 36, p. 283–299, 2015. ISSN 1568-4946. Citado 2 vezes nas páginas 37 e 38.

KOTELNIKOVA, I. Increasing the competitiveness of enterprises under the conditions of digitalization. *Innovation and Sustainability*, 2022. Citado na página 23.

KUMAR, Y. J. et al. Multi document summarization based on news components using fuzzy cross-document relations. *Applied Soft Computing Journal*, Elsevier BV, v. 21, p. 265 – 279, 2014. ISSN 15684946. Citado 5 vezes nas páginas 71, 72, 74, 93 e 109.

LABAZANOVA, S.; BOTSIEVA, E.; PERYAKINA, M. Digital technologies in key sectors of the economy in the context of global competition. *SHS Web of Conferences*, 2023. Citado na página 23.

LANDIS, J. R.; KOCH, G. G. The measurement of observer agreement for categorical data. *biometrics*, JSTOR, p. 159–174, 1977. Citado na página 50.

LAVIE, A.; AGARWAL, A. METEOR: An automatic metric for MT evaluation with high levels of correlation with human judgments. In: CALLISON-BURCH, C. et al. (Ed.). *Proceedings of the Second Workshop on Statistical Machine Translation*. Prague, Czech Republic: Association for Computational Linguistics, 2007. p. 228–231. Citado 6 vezes nas páginas 43, 44, 113, 116, 119 e 129.

LEE, D. D.; SEUNG, H. S. Learning the parts of objects by non-negative matrix factorization. *nature*, Nature Publishing Group UK London, v. 401, n. 6755, p. 788–791, 1999. Citado 3 vezes nas páginas 39, 40 e 41.

LEE, E. K.; UPPAL, K. Cerc: an interactive content extraction, recognition, and construction tool for clinical and biomedical text. *BMC MEDICAL INFORMATICS AND DECISION MAKING*, v. 20, n. 14, SI, DEC 15 2020. Citado 6 vezes nas páginas 59, 63, 64, 67, 68 e 69.

LEVITIN, A. *Introduction to the Design & Analysis of Algorithms*. 3rd. ed. Boston, MA, USA: Pearson, 2012. Includes bibliographical references and index. ISBN 978-0-13-231681-1. Citado na página 138.

LEWIS, M. et al. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *CoRR*, abs/1910.13461, 2019. Citado na página 115.

LIMA, M. S. M.; DELEN, D. Predicting and explaining corruption across countries: A machine learning approach. *Government Information Quarterly*, v. 37, n. 1, p. 101407, 2020. ISSN 0740-624X. Citado 2 vezes nas páginas 36 e 37.

LIN, C.-Y. ROUGE: A package for automatic evaluation of summaries. In: *Text Summarization Branches Out*. Barcelona, Spain: Association for Computational Linguistics, 2004. p. 74–81. Citado 7 vezes nas páginas 43, 77, 96, 113, 116, 119 e 129.

LUO, Z. et al. Towards accurate and clinically meaningful summarization of electronic health record notes: A guided approach. In: IEEE; IEEE ENGN MED & BIOL SOC; IEEE FUTURE DIRECT; NATIL INST HLTH; GOOGLE; NSF; UPMC HILLMAN CANC CTR. *2023 IEEE EMBS INTERNATIONAL CONFERENCE ON BIOMEDICAL AND HEALTH INFORMATICS, BHI*. [S.l.], 2023. (IEEE EMBS International Conference on Biomedical and Health Informatics). ISBN 979-8-3503-1050-4. ISSN 2641-3590. IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI), Pittsburgh, PA, OCT 15-18, 2023. Citado 7 vezes nas páginas 59, 61, 64, 66, 67, 68 e 69.

MACKEY, T. K. et al. The sustainable development goals as a framework to combat health-sector corruption. *Bulletin of The World Health Organization*, 2018. Citado na página 25.

MADUREIRA, L.; POPOVIč, A.; CASTELLI, M. Competitive intelligence: A unified view and modular definition. *Technological Forecasting & Social Change*, Elsevier Inc., v. 173, p. 121086, 2021. ISSN 0040-1625. Received 22 December 2020; Received in revised form 26 July 2021; Accepted 28 July 2021; Available online 9 August 2021; ©2021 Elsevier Inc. All rights reserved. Citado na página 26.

MAMATHA, G.; JOSHI, H. V.; AMITH, R. Chapter eleven - bio-intelligent computing and optimization techniques for developing computerized solutions. In: BISWAS, A. et al. (Ed.). *Applications of Nature-Inspired Computing and Optimization Techniques*. Elsevier, 2024, (Advances in Computers, v. 135). p. 259–288. Disponível em: <https://www.sciencedirect.com/science/article/pii/S006524582300089X>. Citado na página 34.

MASROM, S. et al. Machine learning prediction of petty corruption intention among law enforcement officers. *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, Institute of Advanced Engineering and Science (IAES), v. 30, n. 3, p. 1634–1642, 2023. ISSN 2502-4752. Open Access, Creative Commons Attribution-ShareAlike 4.0 International License. Citado na página 37.

MEHRABI, S. et al. Event causality identification using conditional random field in geriatric care domain. In: *2013 12th International Conference on Machine Learning and Applications*. [S.l.: s.n.], 2013. v. 1, p. 339–343. Citado 7 vezes nas páginas 59, 63, 64, 65, 67, 68 e 69.

MENDOZA, M. et al. Extractive single-document summarization based on genetic operators and guided local search. *Expert Systems with Applications*, v. 41, n. 9, p. 4158 – 4169, 2014. ISSN 09574174. Citado 6 vezes nas páginas 33, 34, 71, 72, 73 e 74.

Meta Team. *Llama 3.2: Revolutionizing edge AI and vision with open, customizable models*. 2024. Citado na página 115.

METRÓPOLES, P. *CGU aponta distorção de R$ 44 bilhões no Ministério da Saúde em 2023*. 2024. Metrópoles. Acesso em: 11 mar. 2026. Disponível em: <https://www.metropoles.com/colunas/paulo-cappelli/cgu-aponta-distorcao-de-r-44-bilhoes-no-ministerio-da-saude>. Citado na página 25.

MISHRA, R. et al. Text summarization in the biomedical domain: a systematic review of recent research. *J. Biomed. Inform.*, v. 52, p. 457–467, dez. 2014. Citado 3 vezes nas páginas 54, 61 e 64.

MOJRIAN, M.; MIRROSHANDEL, S. A. A novel extractive multi-document text summarization system using quantum-inspired genetic algorithm: Mtsqiga. *Expert Systems with Applications*, Elsevier Ltd, v. 171, 2021. ISSN 09574174. Citado 3 vezes nas páginas 71, 72 e 73.

MORID, M. A. et al. Classification of clinically useful sentences in clinical evidence resources. *J. Biomed. Inform.*, Elsevier BV, v. 60, p. 14–22, abr. 2016. Citado 6 vezes nas páginas 59, 63, 64, 67, 68 e 69.

MURAT, A.; SAIDA, S.; DZHAMILYA, K. Social and economic effects of digital transformation. *SHS Web of Conferences*, 2023. Citado na página 23.

NENKOVA, A.; MCKEOWN, K. Automatic summarization. *Foundations and Trends in Information Retrieval*, 2011. Citado na página 115.

NIST. *Proceedings of the Document Understanding Conference*. 2025. Citado 5 vezes nas páginas 77, 87, 96, 104 e 117.

OBI, C. E. et al. Contribution and influence of social capital on corruption in the health sector: A view through the lens of service users. *BMJ Global Health*, v. 10, p. e020195, 2025. Disponível em: <https://doi.org/10.1136/bmjgh-2025-020195>. Citado na página 25.

OZYEGEN, O.; KABE, D.; CEVIK, M. Word-level text highlighting of medical texts for telehealth services. *Artif. Intell. Med.*, Elsevier BV, v. 127, n. 102284, p. 102284, maio 2022. Citado 5 vezes nas páginas 59, 62, 67, 68 e 69.

PAGE, M. J. et al. The prisma 2020 statement: an updated guideline for reporting systematic reviews. *BMJ*, p. n71, 2021. Citado na página 53.

PAPINENI, K. et al. Bleu: a method for automatic evaluation of machine translation. In: ISABELLE, P.; CHARNIAK, E.; LIN, D. (Ed.). *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*. Philadelphia, Pennsylvania, USA: Association for Computational Linguistics, 2002. p. 311–318. Citado 6 vezes nas páginas 43, 44, 113, 116, 119 e 129.

Parsifal. *Parsifal*. 2025. Disponível em: https://parsif.al/. Acesso em: janeiro de 2025. Citado na página 56.

PAULA, T. D. et al. Automated admissibility of complaints about fraud and corruption. In: GAMALLO, P. et al. (Ed.). *Proceedings of the 16th International Conference on Computational Processing of Portuguese - Vol. 1*. Santiago de Compostela, Galicia/Spain: Association for Computational Lingustics, 2024. p. 610–613. Citado na página 26.

PEDREGOSA, F. et al. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, v. 12, p. 2825–2830, 2011. Citado na página 79.

PIRES, R. et al. Sabiá: Portuguese large language models. In: NALDI, M. C.; BIANCHI, R. A. C. (Ed.). *Intelligent Systems*. Cham: Springer Nature Switzerland, 2023. p. 226–240. ISBN 978-3-031-45392-2. Citado na página 115.

PRISMA Executive. *PRISMA 2020 checklist*. 2024. [Accessed 5 December 2024]. Citado na página 53.

Qwen Team. *Qwen2.5: A Party of Foundation Models*. 2024. Citado na página 115.

RABUZIN, K.; MODRUŠAN, N. Prediction of public procurement corruption indices using machine learning methods. In: INSTICC. *Proceedings of the 11th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2019) - KMIS*. [S.l.]: SciTePress, 2019. p. 333–340. ISBN 978-989-758-382-7. ISSN 2184-3228. Citado 2 vezes nas páginas 37 e 38.

REY-PUECH, P. del; BALABANOVA, D.; MCKEE, M. Artificial intelligence and corruption: Opportunities and challenges in the health sector. *The International Journal of Health Planning and Management*, v. 40, n. 6, p. 1341–1347, 2025. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1002/hpm.70002>. Citado na página 25.

RÖDER, M.; BOTH, A.; HINNEBURG, A. Exploring the space of topic coherence measures. In: *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining (WSDM '15)*. New York, NY, USA: ACM, 2015. p. 399–408. ISBN 978-1-4503-3317-7. Citado na página 45.

SAINI, N.; SAHA, S.; BHATTACHARYYA, P. Multiobjective-based approach for microblog summarization. *IEEE Transactions on Computational Social Systems*, Institute of Electrical and Electronics Engineers Inc., v. 6, n. 6, p. 1219 – 1231, 2019. ISSN 2329924X. Citado 4 vezes nas páginas 71, 72, 73 e 74.

SAINI, N. et al. Extractive single document summarization using multi-objective optimization: Exploring self-organized differential evolution, grey wolf optimizer and water cycle algorithm. *Knowledge-Based Systems*, Elsevier B.V., v. 164, p. 45 – 67, 2019. ISSN 09507051. Citado 5 vezes nas páginas 33, 34, 71, 72 e 74.

SAJJA, P. S.; AKERKAR, R. 12 - bio-inspired models for semantic web. In: YANG, X.-S. et al. (Ed.). *Swarm Intelligence and Bio-Inspired Computation*. Oxford: Elsevier, 2013. p. 273–294. ISBN 978-0-12-405163-8. Disponível em: <https://www.sciencedirect.com/science/article/pii/B9780124051638000120>. Citado na página 34.

SALTON, G.; BUCKLEY, C. Term-weighting approaches in automatic text retrieval. *Information Processing & Management*, v. 24, n. 5, p. 513–523, 1988. ISSN 0306-4573. Citado 3 vezes nas páginas 76, 135 e 149.

SANCHEZ-GOMEZ, J. M.; VEGA-RODRíGUEZ, M. A.; PéREZ, C. J. Extractive multi-document text summarization using a multi-objective artificial bee colony optimization approach. *Knowledge-Based Systems*, Elsevier B.V., v. 159, p. 1 – 8, 2018. ISSN 09507051. Citado 4 vezes nas páginas 33, 34, 71 e 72.

SANCHEZ-GOMEZ, J. M.; VEGA-RODRíGUEZ, M. A.; PéREZ, C. J. A decomposition-based multi-objective optimization approach for extractive multi-document text summarization. *Applied Soft Computing Journal*, Elsevier Ltd, v. 91, 2020. ISSN 15684946. Citado 3 vezes nas páginas 33, 71 e 72.

SANCHEZ-GOMEZ, J. M.; VEGA-RODRíGUEZ, M. A.; PéREZ, C. J. A multi-objective memetic algorithm for query-oriented text summarization: Medicine texts as a case study. *Expert Systems with Applications*, v. 198, p. 116769, 2022. ISSN 0957-4174. Citado 6 vezes nas páginas 23, 33, 71, 72, 85 e 101.

SANCHEZ-GOMEZ, J. M.; VEGA-RODRíGUEZ, M. A.; PéREZ, C. J. An indicator-based multi-objective variable neighborhood search approach for query-focused summarization. *Swarm and Evolutionary Computation*, Elsevier B.V., v. 91, 2024. ISSN 22106502. Citado 4 vezes nas páginas 33, 71, 72 e 74.

SANTOS, E. Schneider dos et al. Detection of fraud in public procurement using data-driven methods: a systematic mapping study. *EPJ Data Science*, v. 14, p. 52, 2025. Citado 2 vezes nas páginas 37 e 38.

SARITAS, O. et al. Big data augmentated business trend identification: the case of mobile commerce. *Scientometrics*, 2021. Citado na página 23.

SARKER, A.; MOLLÁ, D.; PARIS, C. Query-oriented evidence extraction to support evidence-based medicine practice. *J. Biomed. Inform.*, Elsevier BV, v. 59, p. 169–184, fev. 2016. Citado 7 vezes nas páginas 59, 63, 64, 65, 67, 68 e 69.

Senado Federal, B. *Orçamento 2025: quase R$ 1 trilhão para Previdência e R$ 245 bilhões para saúde*. [S.l.], 2025. Acesso em: 11 mar. 2026. Disponível em: <https://www12.senado.leg.br/noticias/materias/2025/04/14/orcamento-2025-quase-r-1-trihao-para-previdencia-e-r-245-bilhoes-para-saude>. Citado na página 25.

SHANG, Y. et al. Enhancing biomedical text summarization using semantic relation extraction. *PLoS One*, Public Library of Science (PLoS), v. 6, n. 8, p. e23862, ago. 2011. Citado 7 vezes nas páginas 59, 63, 64, 65, 67, 68 e 69.

SHANNON, C. E. A mathematical theory of communication. *The Bell System Technical Journal*, v. 27, n. 3, p. 379–423, July 1948. ISSN 0005-8580. Citado 2 vezes nas páginas 80 e 100.

SILVA, R.; MAMEDE, H.; SANTOS, A. The role of digital marketing in increasing smes' competitiveness. p. 93–100, 2022. Citado na página 23.

SONNTAG, D.; PROFITLICH, H.-J. Integrated decision support by combining textual information extraction, facetted search and information visualisation. In: *2017 IEEE 30th International Symposium on Computer-Based Medical Systems (CBMS)*. [S.l.: s.n.], 2017. p. 95–100. ISSN 2372-9198. Citado 7 vezes nas páginas 59, 61, 64, 65, 67, 68 e 69.

SOWMYA, R. et al. Data mining with big data. *International Symposium on Combinatorial Optimization*, 2017. Citado na página 24.

STEINBERGER, J.; JEžEK, K. Text summarization and singular value decomposition. In: *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. [S.l.: s.n.], 2004. Citado na página 115.

SWAYAMSIDDHA, S. Chapter 4 - bio-inspired algorithms: principles, implementation, and applications to wireless communication. In: YANG, X.-S. (Ed.). *Nature-Inspired Computation and Swarm Intelligence*. Academic Press, 2020. p. 49–63. ISBN 978-0-12-819714-1. Disponível em: <https://www.sciencedirect.com/science/article/pii/B9780128197141000130>. Citado na página 35.

TANG, V. et al. An adaptive clinical decision support system for serving the elderly with chronic diseases in healthcare industry. *EXPERT SYSTEMS*, v. 36, n. 2, APR 5 2019. ISSN 0266-4720. Citado 7 vezes nas páginas 59, 62, 64, 65, 67, 68 e 69.

TEH, Y. W. et al. Hierarchical dirichlet processes. *Journal of the american statistical association*, Taylor & Francis, v. 101, n. 476, p. 1566–1581, 2006. Citado 3 vezes nas páginas 39, 40 e 42.

TEKNIUM, R.; QUESNELLE, J.; GUANG, C. *Hermes 3 Technical Report*. 2024. Citado na página 115.

TOMER, M.; KUMAR, M. Multi-document extractive text summarization based on firefly algorithm. *Journal of King Saud University - Computer and Information Sciences*, v. 34, n. 8, Part B, p. 6057–6065, 2022. ISSN 1319-1578. Citado 4 vezes nas páginas 71, 72, 93 e 109.

TRAVASSOS, G. H.; GUROV, D.; AMARAL, E. *Introdução à Engenharia de Software*. Rio de Janeiro, RJ, Brasil, 2020. Experimental. Citado na página 47.

VASCONCELOS, M. O.; CHAIM, R. M.; CAVIQUE, L. Imbalanced learning in assessing the risk of corruption in public administration. In: MARREIROS, G. et al. (Ed.). *Progress in Artificial Intelligence*. Cham: Springer International Publishing, 2021. p. 510–523. ISBN 978-3-030-86230-5. Citado na página 37.

VERMA, P.; VERMA, A.; PAL, S. An approach for extractive text summarization using fuzzy evolutionary and clustering algorithms. *Applied Soft Computing*, Elsevier Ltd, v. 120, 2022. ISSN 15684946. Citado 8 vezes nas páginas 71, 72, 73, 74, 75, 93, 109 e 110.

WALTINGER, U. et al. Market intelligence: Linked data-driven entity resolution for customer and competitor analysis. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, v. 7977 LNCS, p. 467 – 481, 2013. ISSN 16113349. Citado 2 vezes nas páginas 24 e 33.

WANG, F. et al. A survey on small language models in the era of large language models: Architecture, capabilities, and trustworthiness. In: *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*. New York, NY, USA: Association for Computing Machinery, 2025. (KDD '25), p. 6173–6183. ISBN 9798400714542. Citado 3 vezes nas páginas 35, 36 e 115.

WANG, M. et al. A systematic review of automatic text summarization for biomedical literature and EHRs. *J. Am. Med. Inform. Assoc.*, Oxford University Press (OUP), v. 28, n. 10, p. 2287–2297, set. 2021. Citado 3 vezes nas páginas 54, 61 e 64.

WANG, X. et al. Open information extraction with meta-pattern discovery in biomedical literature. In: *Proceedings of the 2018 ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics*. New York, NY, USA: Association for Computing Machinery, 2018. (BCB '18), p. 291–300. ISBN 9781450357944. Citado 7 vezes nas páginas 59, 61, 64, 66, 67, 68 e 69.

WEICHSELBRAUN, A. et al. Classifying news media coverage for corruption risks management with deep learning and web intelligence. In: *Proceedings of the 10th International Conference on Web Intelligence, Mining and Semantics*. New York, NY, USA: Association for Computing Machinery, 2020. (WIMS 2020), p. 54–62. ISBN 9781450375429. Citado 2 vezes nas páginas 37 e 38.

WOHLIN, C. et al. *Experimentation in Software Engineering*. 2. ed. Berlin, Heidelberg: Springer, 2024. 274 p. XXV + 274 pages, 1 illustration in colour. ISBN 978-3-662-69305-6. Disponível em: <https://doi.org/10.1007/978-3-662-69306-3>. Citado na página 47.

WOODSEND, K.; LAPATA, M. Learning to simplify sentences with quasi-synchronous grammar and integer programming. In: *EMNLP 2011 - Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*. [S.l.: s.n.], 2011. Citado na página 115.

YANG, R. et al. Ascle-A python natural language processing toolkit for medical text generation: Development and evaluation study. *J. Med. Internet Res.*, JMIR Publications Inc., v. 26, p. e60601, out. 2024. Citado 7 vezes nas páginas 59, 62, 64, 65, 67, 68 e 69.

ZHANG, T. et al. Bertscore: Evaluating text generation with bert. In: *Proceedings of the International Conference on Learning Representations (ICLR)*. [s.n.], 2020. Published as a conference paper at ICLR 2020. Disponível em: <https://arxiv.org/abs/1904.09675>. Citado 6 vezes nas páginas 43, 44, 113, 116, 119 e 129.

ZHU, W.; ZENG, N.; WANG, N. Sensitivity, specificity, accuracy, associated confidence interval and roc analysis with practical sas implementations. In: *NorthEast SAS Users Group, Health Care and Life Sciences*. [S.l.: s.n.], 2010. Citado 2 vezes nas páginas 45 e 143.

ZIRIKLY, A. et al. Information extraction framework for disability determination using a mental functioning use-case. *JMIR MEDICAL INFORMATICS*, v. 10, n. 3, MAR 2022. Citado 6 vezes nas páginas 59, 64, 65, 67, 68 e 69.

APÊNDICE A – Comparação entre Resumos Humanos e Automáticos

Tabela 52 – Comparison between human and automatic summaries.

Human Summary 1
O Conselho Regional de Medicina do Estado de Goiás (Cremego) informou que, até o momento, não foi notificado sobre qualquer ação da Polícia Federal referente às investigações sobre a conduta do médico oftalmologista Farley Vinícius Alcântara, por ele atestar que Gabriela Santiago Ribeiro Cid, esposa do ex-ajudante de ordens de Jair Bolsonaro, Mauro Cid, foi imunizada contra a Covid-19 no município. Segundo as investigações da operação venire, foi relatado que dados da vacina foram falsificados, entre eles, a do ex-presidente Jair Bolsonaro; de sua filha mais nova, Laura Bolsonaro; de Mauro Cid, um dos auxiliares de Bolsonaro presos; da esposa de Cid; e da filha de Cid. Seis pessoas já foram presas e outros 16 mandados foram cumpridos.
Automatic Summaries
NousResearch/Hermes-3-Llama-3.2-3B: O Conselho Regional de Medicina do Estado de Goiás (Cremego) informou que não foi notificado sobre ações da Polícia Federal relacionadas a investigações contra o médico Farley Vinícius Alcântara. O oftalmologista é suspeito de ter fornecido um documento atestando que a mulher do ex-ajudante de ordens de Jair Bolsonaro, Mauro Cid, se imunizou contra a Covid-19 no município de Cabeceiras. A Polícia Federal deflagrou a Operação Venire, apurando que dados de vacina contra a Covid foram falsificados, envolvendo ex-presidente Jair Bolsonaro, sua filha Laura, Mauro Cid, esposa de Cid e filha de Cid. Foram presos seis pessoas e cumpridos 16 mandados. Farley, servidor da Secretaria de Saúde do Distrito Federal, copiou dados de vacinação de uma enfermeira de Cabeceiras para falsificar um cartão de imunização
Qwen/Qwen2.5-7B-Instruct: A Polícia Federal deflagrou a Operação Venire em 3 de maio, investigando fraudes em registros de vacinação contra a COVID-19 envolvendo seis pessoas, incluindo membros da família Bolsonaro e assessores próximos. Farley Vinícius Alcântara, médico oftalmologista, foi acusado de falsificar dados de vacinação para a esposa de Mauro Cid. O crime não foi consumado devido à falta de distribuição de lotes de vacinas para o Rio de Janeiro. Farley é servidor da Secretaria de Saúde do DF, mas trabalha em uma clínica no Rio de Janeiro. O prefeito de Cabeceiras negou que Bolsonaro ou seus assessores se vacinaram no município. O caso está sendo investigado e pode resultar em processos administrativos contra o médico.

meta-llama/Llama-3.2-3B-Instruct: Aqui está um resumo objetivo e conciso do conteúdo em português: A Polícia Federal (PF) deflagrou a Operação Venire, que envolveu a investigação de um esquema de fraude em registros de vacinação contra a Covid-19. O médico Farley Vinícius Alcântara, que trabalha em uma clínica particular de olhos no Rio de Janeiro, foi acusado de copiar dados de vacinação de uma enfermeira de Cabeceiras, interior de Goiás, para falsificar um cartão de imunização para a esposa de Mauro Cid, um dos auxiliares de Jair Bolsonaro presos. O prefeito de Cabeceiras, Everton Francisco de Matos (PDT), conhecido como Tuta, afirmou que nenhum membro da família de Bolsonaro ou qualquer assessor se vacinou no município. Ele também afirmou que o município está isento disso e que a atitude do médico Farley está prejudicando

Human Summary 2

O ministro da Controladoria-Geral da União, Wagner Rosário, afirmou à CNN que a CGU abriu investigação sobre o caso Davati. A apuração busca esclarecer o encontro entre o empresário Luiz Paulo Dominguetti e o ex-diretor do Ministério da Saúde, Roberto Dias, onde teria havido pedido de propina de US$ 1 por dose de vacina. Segundo Rosário, o sistema do Ministério registrou proposta de 300 milhões de doses pela Davati, mas sem vínculo oficial com Dominguetti. Ele também confirmou a entrada do empresário no ministério em março de 2021. O ministro disse que, até o momento, não há irregularidades no contrato da vacina Covaxin, mas apontou que houve três problemas no invoice, um tipo de nota fiscal, mas que todos já foram corrigidos. Rosário declarou estar pronto para depor na CPI da Pandemia, se convocado, e evitou comentar as críticas do presidente Jair Bolsonaro à comissão.

Automatic Summaries

NousResearch/Hermes-3-Llama-3.2-3B: Em entrevista exclusiva à CNN, o ministro-chefe da Controladoria-Geral da União, Wagner Rosário, afirmou que a CGU abriu uma investigação para apurar o caso Davati e uma suposta cobrança de propina. A CGU quer entender melhor o que aconteceu no encontro entre o empresário Luiz Paulo Dominguetti Pereira e o então diretor do Ministério da Saúde Roberto Ferreira Dias, em um restaurante de um shopping de Brasília. A CGU também apurou que Dominguetti esteve no prédio do Ministério da Saúde no dia 26 de março de 2021, com entrada às 13h39. Rosário, porém, não soube informar com quem o empresário esteve. Wagner Rosário disse que "processualmente"não há irregularidades nos contratos de contratação e importação da vacina indiana Covaxin, mas o relatório ainda não está concluído. O departamento do Ministério da Saúde que cuida dessa área identificou três

Qwen/Qwen2.5-7B-Instruct: Aqui está uma versão concisa e objetiva do texto em português: O ministro-chefe da Controladoria-Geral da União, Wagner Rosário, concedeu uma entrevista à CNN sobre investigações relacionadas à pandemia de COVID-19 no Brasil. Ele afirmou que a CGU está investigando um suposto pedido de propina e um encontro entre um empresário e um ex-dirigente do Ministério da Saúde. Rosário também declarou que não há irregularidades processuais nas contratações da vacina Covaxin, apesar de problemas encontrados no processo de importação das vacinas. Ele indicou que esses problemas foram corrigidos antes da conclusão do contrato. O ministro admitiu a possibilidade de depor à CPI da Pandemia, caso seja convocado pelos senadores. Ele também comentou sobre críticas feitas pelo presidente Jair Bolsonaro à CPI, afirmando que cada cidadão deve avaliar o comportamento dos membros da comissão. Resumindo, a principal informação é que a CGU está investig

meta-llama/Llama-3.2-3B-Instruct: Aqui está um resumo objetivo e conciso do conteúdo em português: O ministro-chefe da Controladoria-Geral da União, Wagner Rosário, afirmou que a CGU abriu uma investigação para apurar o caso Davati e uma suposta cobrança de propina. Ele também admitiu que está pronto para depor à CPI da Pandemia, caso seja acionado pelos senadores que integram a comissão. Rosário ressaltou que o relatório ainda não está concluído e, por isso, não poderia se manifestar definitivamente sobre o caso. Ele também afirmou que não há irregularidades nos contratos de contratação e importação da vacina indiana Covaxin. A CGU apurou que o empresário Luiz Paulo Dominguetti Pereira teria recebido um pedido de propina de US$ 1 por dose de vacina da Astrazeneca em troca de fechar contrato com o Ministério da Saúde. No entanto, Rosário não soube informar com quem o empresário esteve

APÊNDICE B – Teste Wilcox Singed-Rank Pairwise

Tabela 53 – Pairwise p-value for ROUGE-1 F1 scores among evaluated models.

Model	Hermes-3-Llama-3.2-3B	Qwen2.5-0.5B	Qwen2.5-1.5B	Qwen2.5-7B	Tucano-1b1	Tucano-2b4	gemma-2b-it	klsum	lexrank	lsa	sabia-7b	Llama-3.2-3B	TeenyTinyLlama	sumbasic	textrank
Hermes-3-Llama-3.2-3B	NaN	0.0	0.0	0.792063	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.066455	0.0	0.0	0.0
Qwen2.5-0.5B	1.0	NaN	1.0	1.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	1.0	0.0
Qwen2.5-1.5B	1.0	0.0	NaN	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	1.0	0.0
Qwen2.5-7B	0.207937	0.0	0.0	NaN	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.009585	0.0	0.0	0.0
Tucano-1b1	1.0	1.0	1.0	1.0	NaN	1.0	0.0	0.0	0.817954	0.0	0.0	1.0	0.0	1.0	0.0
Tucano-2b4	1.0	0.0	1.0	1.0	0.0	NaN	0.0	0.0	0.0	0.0	0.0	1.0	0.0	1.0	0.0
gemma-2b-it	1.0	1.0	1.0	1.0	1.0	1.0	NaN	0.0	1.0	0.0	0.0	1.0	0.0	1.0	0.0
klsum	1.0	1.0	1.0	1.0	1.0	1.0	1.0	NaN	1.0	1.0	0.0	1.0	1.0	1.0	0.0
lexrank	1.0	1.0	1.0	1.0	0.182046	1.0	0.0	0.0	NaN	0.0	0.0	1.0	0.0	1.0	0.0
lsa	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.0	1.0	NaN	0.0	1.0	0.0	1.0	0.0
sabia-7b	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	NaN	1.0	1.0	1.0	0.999998
Llama-3.2-3B	0.933545	0.0	0.0	0.990415	0.0	0.0	0.0	0.0	0.0	0.0	0.0	NaN	0.0	0.0	0.0
TeenyTinyLlama	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.0	1.0	1.0	0.0	1.0	NaN	1.0	0.0
sumbasic	1.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	NaN	0.0
textrank	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.000002	1.0	1.0	1.0	NaN

Tabela 54 – Pairwise p-value for ROUGE-2 F1 scores among evaluated models.

Model	Hermes-3-Llama-3.2-3B	Qwen2.5-0.5B	Qwen2.5-1.5B	Qwen2.5-7B	Tucano-1b1	Tucano-2b4	gemma-2b-it	klsum	lexrank	lsa	sabia-7b	Llama-3.2-3B	TeenyTinyLlama	sumbasic	textrank
Hermes-3-Llama-3.2-3B	NaN	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Qwen2.5-0.5B	1.0	NaN	1.0	1.0	0.0	1.0	1.0	0.0	1.0	0.0	0.0	1.0	0.0	1.0	0.0
Qwen2.5-1.5B	1.0	0.0	NaN	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	1.0	0.0
Qwen2.5-7B	1.0	0.0	0.0	NaN	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.007489	0.0	0.0	0.0
Tucano-1b1	1.0	1.0	1.0	1.0	NaN	1.0	1.0	0.0	1.0	1.0	0.0	1.0	0.0	1.0	0.0
Tucano-2b4	1.0	0.0	1.0	1.0	0.0	NaN	0.981377	0.0	1.0	0.0	0.0	1.0	0.0	1.0	0.0
gemma-2b-it	1.0	0.0	1.0	1.0	0.0	0.018623	NaN	0.0	1.0	0.0	0.0	1.0	0.0	1.0	0.0
klsum	1.0	1.0	1.0	1.0	1.0	1.0	1.0	NaN	1.0	1.0	1.0	1.0	0.0	1.0	0.999992
lexrank	1.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	NaN	0.0	0.0	1.0	0.0	1.0	0.0
lsa	1.0	1.0	1.0	1.0	0.0	1.0	1.0	0.0	1.0	NaN	0.0	1.0	0.0	1.0	0.0
sabia-7b	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.0	1.0	1.0	NaN	1.0	0.0	1.0	0.000004
Llama-3.2-3B	1.0	0.0	0.0	0.992511	0.0	0.0	0.0	0.0	0.0	0.0	0.0	NaN	0.0	0.0	0.0
TeenyTinyLlama	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	NaN	1.0	1.0
sumbasic	1.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	NaN	0.0
textrank	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.000008	1.0	1.0	0.999996	1.0	0.0	1.0	NaN

Tabela 55 – Pairwise p-value for ROUGE-L scores among evaluated models.

Model	Hermes-3-Llama-3.2-3B	Qwen2.5-0.5B	Qwen2.5-1.5B	Qwen2.5-7B	Tucano-1b1	Tucano-2b4	gemma-2b-it	klsum	lexrank	lsa	sabia-7b	Llama-3.2-3B	TeenyTinyLlama	sumbasic	textrank
NousResearch/Hermes-3-Llama-3.2-3B	NaN	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Qwen/Qwen2.5-0.5B-Instruct	1.0	NaN	1.0	1.0	0.0	1.0	0.999955	0.0	1.0	1.0	0.0	1.0	0.0	1.0	0.0
Qwen/Qwen2.5-1.5B-Instruct	1.0	0.0	NaN	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	1.0	0.0
Qwen/Qwen2.5-7B-Instruct	1.0	0.0	0.0	NaN	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
TucanoBR/Tucano-1b1-Instruct	1.0	1.0	1.0	1.0	NaN	1.0	1.0	0.0	1.0	1.0	0.0	1.0	0.0	1.0	0.0
TucanoBR/Tucano-2b4-Instruct	1.0	0.0	1.0	1.0	0.0	NaN	0.115308	0.0	1.0	0.16874	0.0	1.0	0.0	1.0	0.0
google/gemma-2b-it	1.0	0.000045	1.0	1.0	0.0	0.884692	NaN	0.0	1.0	0.92483	0.0	1.0	0.0	1.0	0.0
klsum	1.0	1.0	1.0	1.0	1.0	1.0	1.0	NaN	1.0	1.0	0.0	1.0	0.0	1.0	0.985231
lexrank	1.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	NaN	0.0	0.0	1.0	0.0	1.0	0.0
lsa	1.0	0.0	1.0	1.0	0.0	0.83126	0.07517	0.0	1.0	NaN	0.0	1.0	0.0	1.0	0.0
maritaca-ai/sabia-7b	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	NaN	1.0	1.0	1.0	1.0
meta-llama/Llama-3.2-3B-Instruct	1.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	NaN	0.0	0.0	0.0
nicholasKluge/TeenyTinyLlama-460m-Chat	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.0	1.0	NaN	1.0	1.0
sumbasic	1.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	NaN	0.0
textrank	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.014769	1.0	1.0	0.0	1.0	0.0	1.0	NaN

Tabela 56 – Pairwise p-value for BLEU scores among evaluated models.

Model	Hermes-3-Llama-3.2-3B	Qwen2.5-0.5B	Qwen2.5-1.5B	Qwen2.5-7B	Tucano-1b1	Tucano-2b4	gemma-2b-it	klsum	lexrank	lsa	sabia-7b	Llama-3.2-3B	TeenyTinyLlama	sumbasic	textrank
Hermes-3-Llama-3.2-3B	NaN	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Qwen2.5-0.5B-Instruct	1.0	NaN	1.0	1.0	0.0	0.999995	1.0	0.0	1.0	0.000001	0.0	1.0	0.0	1.0	0.0
Qwen2.5-1.5B-Instruct	1.0	0.0	NaN	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	1.0	0.0
Qwen2.5-7B-Instruct	1.0	0.0	0.0	NaN	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.995113	0.0	0.0	0.0
Tucano-1b1-Instruct	1.0	1.0	1.0	1.0	NaN	1.0	1.0	0.0	1.0	1.0	0.0	1.0	0.0	1.0	0.0
Tucano-2b4-Instruct	1.0	0.000005	1.0	1.0	0.0	NaN	1.0	0.0	1.0	0.0	0.0	1.0	0.0	1.0	0.0
gemma-2b-it	1.0	0.0	1.0	1.0	0.0	0.0	NaN	0.0	0.999549	0.0	0.0	1.0	0.0	1.0	0.0
klsum	1.0	1.0	1.0	1.0	1.0	1.0	1.0	NaN	1.0	1.0	0.999847	1.0	0.0	1.0	0.999294
lexrank	1.0	0.0	1.0	1.0	0.0	0.0	0.000451	0.0	NaN	0.0	0.0	1.0	0.0	1.0	0.0
lsa	1.0	0.999999	1.0	1.0	0.0	1.0	1.0	0.0	1.0	NaN	0.0	1.0	0.0	1.0	0.0
sabia-7b	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.000153	1.0	1.0	NaN	1.0	0.0	1.0	0.1481
Llama-3.2-3B-Instruct	1.0	0.0	0.0	0.004887	0.0	0.0	0.0	0.0	0.0	0.0	0.0	NaN	0.0	0.0	0.0
TeenyTinyLlama-460m-Chat	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	NaN	1.0	1.0
sumbasic	1.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	NaN	0.0
textrank	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.000706	1.0	1.0	0.8519	1.0	0.0	1.0	NaN

Tabela 57 – Pairwise p-value for METEOR scores among evaluated models.

Model	Hermes-3-Llama-3.2-3B	Qwen2.5-0.5B	Qwen2.5-1.5B	Qwen2.5-7B	Tucano-1b1	Tucano-2b4	gemma-2b-it	klsum	lexrank	lsa	sabia-7b	Llama-3.2-3B	TeenyTinyLlama	sumbasic	textrank
Hermes-3-Llama-3.2-3B	NaN	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000462	0.0	0.0	0.0
Qwen2.5-0.5B	1.0	NaN	1.0	1.0	0.0	1.0	0.013323	0.0	1.0	1.0	0.0	1.0	0.0	1.0	0.0
Qwen2.5-1.5B	1.0	0.0	NaN	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	1.0	0.0
Qwen2.5-7B	1.0	0.0	0.0	NaN	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
Tucano-1b1	1.0	1.0	1.0	1.0	NaN	1.0	1.0	0.0	1.0	1.0	0.0	1.0	0.0	1.0	0.06444
Tucano-2b4	1.0	0.0	1.0	1.0	0.0	NaN	0.0	0.0	1.0	0.862053	0.0	1.0	0.0	1.0	0.0
gemma-2b-it	1.0	0.986677	1.0	1.0	0.0	1.0	NaN	0.0	1.0	1.0	0.0	1.0	0.0	1.0	0.0
klsum	1.0	1.0	1.0	1.0	1.0	1.0	1.0	NaN	1.0	1.0	0.0	1.0	1.0	1.0	1.0
lexrank	1.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	NaN	0.0	0.0	1.0	0.0	1.0	0.0
lsa	1.0	0.0	1.0	1.0	0.0	0.137947	0.0	0.0	1.0	NaN	0.0	1.0	0.0	1.0	0.0
sabia-7b	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	NaN	1.0	1.0	1.0	1.0
Llama-3.2-3B	0.999538	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	NaN	0.0	0.0	0.0
TeenyTinyLlama	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.0	1.0	1.0	0.0	1.0	NaN	1.0	1.0
sumbasic	1.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	NaN	0.0
textrank	1.0	1.0	1.0	1.0	0.93556	1.0	1.0	0.0	1.0	1.0	0.0	1.0	0.0	1.0	NaN

Tabela 58 – Pairwise p-value for BERTScore F1 scores among evaluated models.

Model	Hermes-3-Llama-3.2-3B	Qwen2.5-0.5B	Qwen2.5-1.5B	Qwen2.5-7B	Tucano-1b1	Tucano-2b4	gemma-2b-it	klsum	lexrank	lsa	sabia-7b	Llama-3.2-3B	TeenyTinyLlama	sumbasic	textrank
Hermes-3-Llama-3.2-3B	NaN	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.449384	0.0	0.0	0.0
Qwen2.5-0.5B	1.0	NaN	1.0	1.0	0.0	1.0	0.0	0.0	1.0	0.0	0.0	1.0	0.0	1.0	0.0
Qwen2.5-1.5B	1.0	0.0	NaN	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	1.0	0.0
Qwen2.5-7B	0.0	0.0	0.0	NaN	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Tucano-1b1	1.0	1.0	1.0	1.0	NaN	1.0	0.0	0.0	1.0	0.0	0.0	1.0	0.0	1.0	0.0
Tucano-2b4	1.0	0.0	1.0	1.0	0.0	NaN	0.0	0.0	0.0	0.0	0.0	1.0	0.0	1.0	0.0
gemma-2b-it	1.0	1.0	1.0	1.0	1.0	1.0	NaN	0.0	1.0	1.0	0.0	1.0	0.024403	1.0	0.000063
klsum	1.0	1.0	1.0	1.0	1.0	1.0	1.0	NaN	1.0	1.0	0.0	1.0	1.0	1.0	1.0
lexrank	1.0	0.0	1.0	1.0	0.0	1.0	0.0	0.0	NaN	0.0	0.0	1.0	0.0	1.0	0.0
lsa	1.0	1.0	1.0	1.0	1.0	1.0	0.0	0.0	1.0	NaN	0.0	1.0	0.0	1.0	0.0
sabia-7b	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	NaN	1.0	1.0	1.0	1.0
Llama-3.2-3B	0.550616	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	NaN	0.0	0.0	0.0
TeenyTinyLlama	1.0	1.0	1.0	1.0	1.0	1.0	0.975597	0.0	1.0	1.0	0.0	1.0	NaN	1.0	0.0
sumbasic	1.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	NaN	0.0
textrank	1.0	1.0	1.0	1.0	1.0	1.0	0.999937	0.0	1.0	1.0	0.0	1.0	1.0	1.0	NaN