Revista
DESENVOLVIMENTO SOCIOECONÔMICO EM DEBATE
e-ISSN: 2446-5496

ARTIGOS LIVRES

Sumarização extrativa automática de artigos médico-científicos referentes a Covid-19 no Brasil

Automatic extractive summary of medical-scientific articles about Covid-19 in Brazil

Daniel Augusto Veronezi Salvador

Bacharel em Ciência da Computação, Universidade do Extremo Sul Catarinense. E-mail: dvsalvador.ds@gmail.com

0009-0008-1251-6078

Érica da Silva Sipriano

Licenciada em Matemática, Universidade do Extremo Sul Catarinense. E-mail: ericassipriano@gmail.com

0000-0002-1301-225X

Kristian Madeira

Doutor em Ciências da Saúde, Universidade do Extremo Sul Catarinense. E-mail: kristian@unesc.net

0000-0002-0929-9403

Merisandra Côrtes de Mattos

Doutora em Engenharia Elétrica, Universidade do Extremo Sul Catarinense. E-mail: mem@unesc.net

0000-0002-7028-8025

Resumo

Sumarização automática de textos utiliza técnicas de processamento de linguagem natural para auxiliar na obtenção de informações relevantes de um texto. A sumarização automática de artigos médico-científicos sobre a Covid-19 no Brasil foi realizada utilizando os algoritmos de Viterbi para marcação discursiva, o K-Means para agrupamento por tópicos e o Term Frequency–Inverse Document Frequency para avaliar a importância de palavras e frases.

Palavras-chave: inteligência artificial; processamento de linguagem natural; covid-19.

Abstract

Automatic text summarization uses natural language processing techniques to help on relevant information retrieval from texts. The automatic summarization of medical-scientific articles on Covid-19 in Brazil, using the Viterbi algorithm for discourse tagging, K-Means for topic clustering, and the Term Frequency–Inverse Document Frequency measure to assess word and phrase importance.

Keywords: artificial inteligence; natural language processing; covid-19.

DOI: 10.18616/rdsd.v10i2.9455

Recebido: 08/06/2024
Aprovado: 25/11/2024

1. Introdução

Comparado à medicina, física e química, a inteligência artificial é um campo de estudos recente, cujo objetivo é compreender o pensamento humano e, além disso, arquitetar entidades similares (Russel; Norvig, 2013). No contexto medicinal, a inteligência artificial tem demonstrado, por exemplo, progresso na diagnose, tratamento e prognose de doenças, por meio de suporte a decisões clínicas e técnicas que possibilitam a avaliação de pacientes. Ela também possibilita a criação de sistemas que podem alcançar desempenho próximo ou superior ao de profissionais especialistas (Hosny; Aerts, 2019).

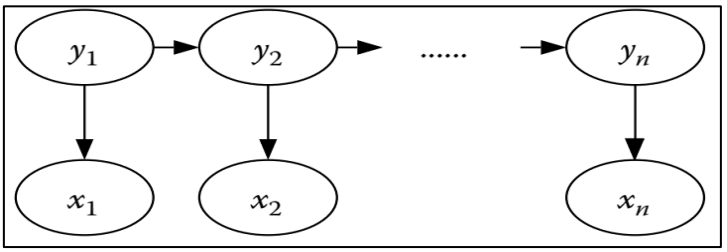
Diversas subáreas existem no campo da inteligência artificial, sendo o Processamento de Linguagem Natural (PLN) dedicado a tarefas relacionadas à interpretação e geração de conteúdo em linguagem escrita ou falada pelos humanos (Hirschberg; Manning, 2015). Há quatro abordagens básicas para a subárea de PLN: simbólica, estatística, conexionista e híbrida.

A abordagem simbólica analisa o fenômeno linguístico baseado em esquemas de representação e algoritmos de associação bem definidos sobre a linguagem, não tolerando ruídos nos dados. A abordagem estatística utiliza diversas técnicas matemáticas e frequentemente grandes quantidades de dados para geração de modelos genéricos aproximados, esta abordagem é muito utilizada para tradução automática, marcação gramatical, reconhecimento de discursos, entre outras tarefas. A abordagem conexionista utiliza uma rede de unidades interconectadas. Similar à estatística, gera modelos genéricos aproximados, porém o que as diferencia é que a abordagem conexionista combina aprendizagem estatística com teorias de representação, permitindo a transformação, inferência e manipulação das fórmulas. Por fim, há também a abordagem híbrida, que emprega duas ou mais abordagens (Russel; Norvig, 2013; Vajjala *et al.*, 2020).

A abordagem estatística tem como objetivo criar modelos probabilísticos a partir de dados variados e desestruturados. Quando utilizada com dados textuais, ela pode ser utilizada para catalogação e extração de informações e, por ser probabilística, ela é tolerável à ruídos nos textos. Alguns algoritmos empregados nesta problemática são o Naive Bayes, o modelo oculto de Markov, do inglês *Hidden Markov Model* (HMM), e a máquina de vetores de suporte (Russel; Norvig, 2013; Vajjala *et al.*, 2020).

O HMM é um modelo gerativo que apresenta estados ocultos. Cada estado é dependente do anterior e cada variável é dependente do estado, portanto, não há informações sobre o estado posterior e os dados somente são observados após a mudança do mesmo, figura 1 (Indurkhya; Damerau, 2010; Manning; Schûtze, 1999; Russel; Norvig, 2013; Vajjala *et al.*, 2020).

Figura 1: Representação gráfica do HMM



Fonte: Indurkhya e Damerau (2010).

Uma das questões fundamentais relacionadas ao HMM se refere a como escolher a melhor sequência de estados para explicar os dados observados. Essa questão pode ser abordada com o algoritmo de Viterbi, cujo modelo computa a melhor sequência de estados. Em cada nó de um grafo de estados é armazenada a probabilidade do caminho mais provável para ele. Ademais, também é registrado o nó que o levou para aquele caminho. Ao final, por meio de programação dinâmica, o melhor caminho é calculado, retrocedendo do melhor estado final para o inicial (Manning; Schûtze, 1999; Russel; Norvig, 2013).

O algoritmo de Viterbi pode ser utilizado para marcação de parte do discurso, a qual é uma etapa importante para aplicações de extração de informações. Dentre elas, existe a tarefa de sumarização automática de textos, que tem como objetivo criar um resumo dos textos. Ela pode ocorrer de forma extrativa ou abstrativa, focada em consulta ou independente, e em documentos únicos ou em diversos documentos. A extrativa, utilizada nesta pesquisa, captura partes do texto para criar um resumo (Vajjala *et al.*, 2020, tradução nossa).

Existem diversos métodos para definir quais partes dos textos devem estar contidas nos resumos, como por exemplo, por meio da importância de uma palavra em relação a outras da mesma sentença e texto, sendo chamada de *Term Frequency–Inverse Document Frequency* (TF-IDF). Este método é criado a partir da multiplicação de duas medidas: *Term Frequency* (TF), apresentada na equação 1, que mede o quão frequente uma palavra é no contexto; e *Inverse Document Frequency* (IDF), demonstrada pela equação 2, que mede o quão raras as palavras são no contexto (Manning; Schûtze, 1999; Vajjala *et al.*, 2020).

$$\text{TF}(td) = \frac{\text{ocorrência do termo } t \text{ no texto } d}{\text{total de palavras no texto } d} \quad (1)$$

$$\text{IDF(t)} = \log_e\left(\frac{\text{número de textos}}{\text{número de textos contendo o termo } t}\right) \quad (2)$$

Um engajamento global, em busca de vacinas, medicamentos e diagnósticos, iniciou a partir da pandemia da Covid-19 (Lopes, 2020). Diversas pesquisas foram realizadas em um curto espaço de tempo, gerando grandes quantidades de resultados (Pham *et al.*, 2020).

Repositórios, como o *Covid-19 Open Research Dataset Challenge* (CORD-19), criado pelo *Allen Institute For Artificial Intelligence*, foram criados para agrupar esses documentos. Contudo, para que essa massiva quantidade de dados seja processada de forma eficiente, é necessária a utilização de técnicas que abordem essas situações, como a inteligência artificial (Neill, 2012).

Nesta pesquisa, com o objetivo de compilar os artigos médico-científicos referentes à Covid-19 no Brasil, observou-se a eficácia e os efeitos colaterais da hidroxicloroquina, para isso, foi desenvolvido um modelo de sumarização automática, extrativa, independente e de documentos únicos, de textos do repositório CORD-19. As técnicas utilizadas para tal foram: o algoritmo de Viterbi, para pré-processamento dos dados; e a medida de distribuição de frequência TF-IDF, para classificação das sentenças. Com base nos percentuais dos experimentos, os artigos foram automaticamente sumarizados e validados por meio do conjunto de métricas *Recall-Oriented Understudy for Gisting Evaluation* (ROUGE), e os resultados comparados com o modelo TextRank.

2. Trabalhos Correlatos

Diversos trabalhos, nacionais e internacionais, sobre sumarização automática de textos foram publicados em diversos periódicos, como por exemplo, os disponibilizados pela IEEE Xplore e ScienceDirect. A pesquisa realizada por Zaware *et al.* (2021) teve como objetivo demonstrar o desempenho da união entre os métodos TF-IDF e TextRank, para sumarização de textos. Para isso, foi utilizado o repositório *BBC News Summary*. Primeiramente, os textos passaram pelas etapas de pré-processamento e normalização dos dados. Após, foi calculada a importância das palavras por meio de TF-IDF. A matriz resultante foi utilizada para calcular a matriz de similaridade de cosseno, o que possibilitou a geração de um grafo para classificação das sentenças com TextRank. Selecionando as sentenças com base em suas pontuações, os sumários foram criados. Foram utilizados os conjuntos de métricas ROUGE-1, ROUGE-2 e ROUGE-L, em comparação com referências próprias para validação, disponibilizadas pelo repositório, para validá-los. Os resultados da sumarização com TF-IDF e TextRank foram comparados com a sumarização utilizando apenas TF-IDF, que indicaram que o modelo proposto gerou resultados superiores ao do modelo isolado.

Rai *et al.* (2021) desenvolveram um modelo para sumarização de artigos do CORD-19, no qual os resumos foram divididos em cinco partes: resultados, tipo do estudo, tamanho da amostra, tipo da amostra e qualidade dos resultados. Em cada parte do resumo, o usuário informou parâmetros para guiar a geração dele. Os dados foram rotulados com *Bidirectional Encoder Representations from Transformers* (BERT), as entidades extraídas com a biblioteca spaCy e as matrizes de posições das palavras foram geradas com a ferramenta pré-treinada para InferSent do Facebook. Cada divisão do resumo possui uma métrica para definir o que

é importante para ela. Ao final, por meio de cálculos de proximidade, as sentenças foram selecionadas e o resumo foi gerado. Para validação do mesmo, foram utilizados os conjuntos de métricas ROUGE-1, ROUGE-2 e ROUGE-L, e os resumos dos artigos como referências, em comparação com o desempenho de outros modelos, como o TextRank e *Latent Semantic Analysis* (LSA). As métricas indicam que o modelo da pesquisa gerou melhores resultados que os modelos comparados.

Rani e Lobiyal (2021) aplicaram TF-IDF, para medir a importância das palavras, e um algoritmo para discernir sentenças semanticamente similares nos textos da *Document Understanding Conference* (DUC), a fim de gerar uma diversidade de resumos, com 5%, 15%, 25% e 50% do texto original. Validando os resultados com ROUGE-1, ROUGE-2 e ROUGE-L, de encontro com referências geradas por especialistas do Instituto Nacional de Padrões e Tecnologias e em comparação com outros métodos conceituados, como TextRank, LexRank e LSA, obtiveram-se valores comprovando que o método da pesquisa apresenta melhor performance que os métodos já conceituados.

Jain, Bellaney e Jangid (2021) realizaram uma pesquisa na qual aplicaram TF-IDF, em conjunto com o banco de palavras ChEMBL e o gerado pelo BERT-*Based Exhaustive Neural Named Entity Recognition and Disambiguation* (BENNERD), para sumarização de artigos aleatórios do repositório CORD-19. Os textos foram pré-processados, lematizados e, após, foi calculada a importância de cada palavra com TF-IDF, recebendo maior importância as que estavam presentes nos bancos de palavras. Utilizando *Agglomerative Nesting* (AGNES), com similaridade de cosseno, as sentenças foram selecionadas gerando agrupamentos com até 25% do texto original. Os resumos foram validados com os conjuntos de métricas ROUGE-1, ROUGE-2 e ROUGE-L, utilizando o resumo dos próprios artigos como referências, cujos resultados foram comparados com o desempenho de ferramentas online de sumarização. O modelo proposto apresentou métricas superiores à de ferramentas online.

Gupta e Patel (2021) utilizaram LSA, com *Singular Value Decomposition* (SVD), TF-IDF e BERT, a fim de sumarizar diferentes textos disponibilizados pelos repositórios de sumarização de textos do Kaggle. Após pré-processar e normalizar os dados, os tópicos foram extraídos com LSA e SVD truncado e as palavras-chaves com TF-IDF. Ambos, então, passaram por BERT para gerar as matrizes de posições das palavras. Ao fim, a pontuação das sentenças foi calculada com similaridade de cosseno e as com as cinco maiores pontuações foram selecionadas. Por meio dos conjuntos de métricas ROUGE-1 e ROUGE-L, utilizando as referências dos repositórios, os sumários foram validados e comparados com um modelo de sumarização que utiliza modelação de tópicos com *Latent Dirichlet Architecture* (LDA). De acordo com as métricas, o modelo proposto teve um melhor desempenho em relação ao modelo com LDA.

O quadro 1 apresenta um comparativo dos trabalhos e da presente pesquisa.

Quadro 1: Comparativo entre os trabalhos

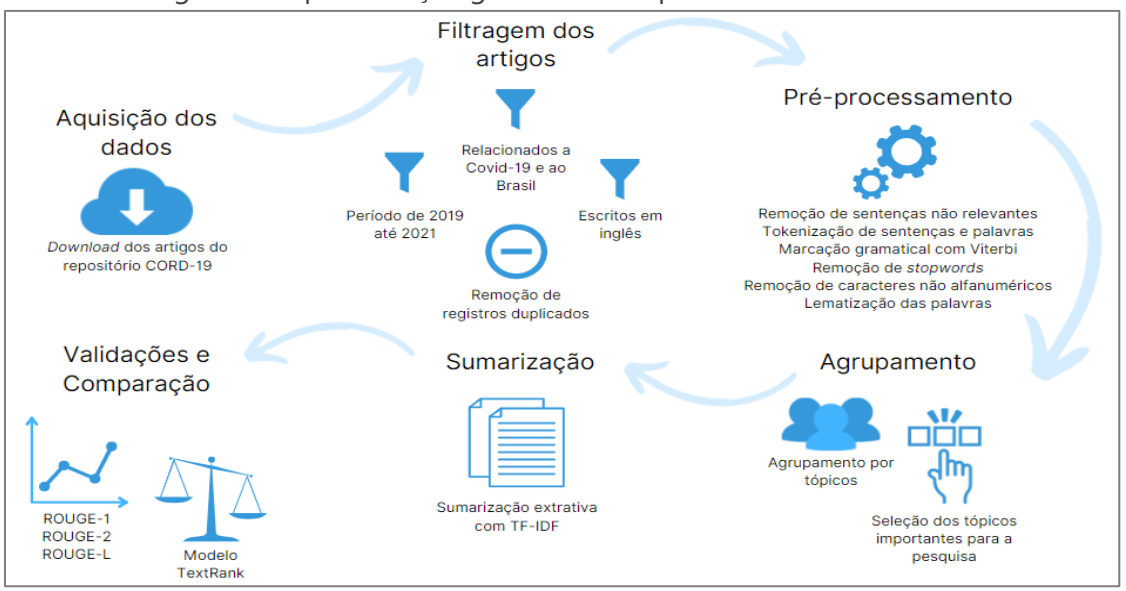
Trabalho	Dados	Resumos Referências	Estratégia	Métricas	Resultados
Zaware *et al.* (2021)	BBC News Summary	Disponibilizados pelo repositório	TF-IDF e TextRank	ROUGE-1 ROUGE-2 ROUGE-L	Resultados superiores ao TF-IDF isolado
Rai *et al.* (2021)	CORD-19	Resumos dos artigos	BERT, spaCy, InferSent e cálculos de proximidade	ROUGE-1 ROUGE-2 ROUGE-L	Resultados superiores ao TextRank e LSA
Rani e Lobiyal (2021)	DUC	Referências geradas por especialistas	TF-IDF, K-Means e algoritmo de classificação de sentenças	ROUGE-1 ROUGE-2 ROUGE-L	Performance melhor que os métodos já conceituados
Jain, Bellaney e Jangid (2021)	CORD-19	Resumos dos artigos	TF-IDF, ChEMBL, BENNERD, AGNES e similaridade de cosseno	ROUGE-1 ROUGE-2 ROUGE-L	Métricas superiores à de ferramentas online
Gupta e Patel (2021)	Repositórios do Kaggle	Disponibilizados pelo repositório	LSA com SVD truncado, TF-IDF, BERT e similaridade de cosseno	ROUGE-1 ROUGE-L	Melhor desempenho em relação ao modelo com LDA
Esta pesquisa	CORD-19	Resumos dos artigos	Viterbi, K-Means e TF-IDF	ROUGE-1 ROUGE-2 ROUGE-L	-

Fonte: Dos autores.

3. Procedimentos metodológicos

A partir desta pesquisa, que é de categoria quantitativa, aplicada, de base tecnológica, descritiva e transversal (Appolinário, 2012), foram aplicados conceitos da abordagem estatística de PLN para sumarização extrativa de textos. Ela objetivou utilizar o algoritmo de Viterbi e classificação da importância das palavras, por meio da medida TF-IDF, para resumir artigos médico-científicos referentes a Covid-19 no Brasil. Com o intuito de selecionar os dados relevantes para esta pesquisa, foram aplicadas técnicas manuais e automáticas de filtragem e agrupamento de dados. A figura 2 apresenta as etapas realizadas para o desenvolvimento desta pesquisa, sendo elas: aquisição dos dados, filtragem dos que eram relevantes para o contexto estudado, pré-processamento, agrupamento e seleção dos tópicos e, por fim, a sumarização extrativa deles.

Figura 2: Representação gráfica das etapas de desenvolvimento



Fonte: Dos autores.

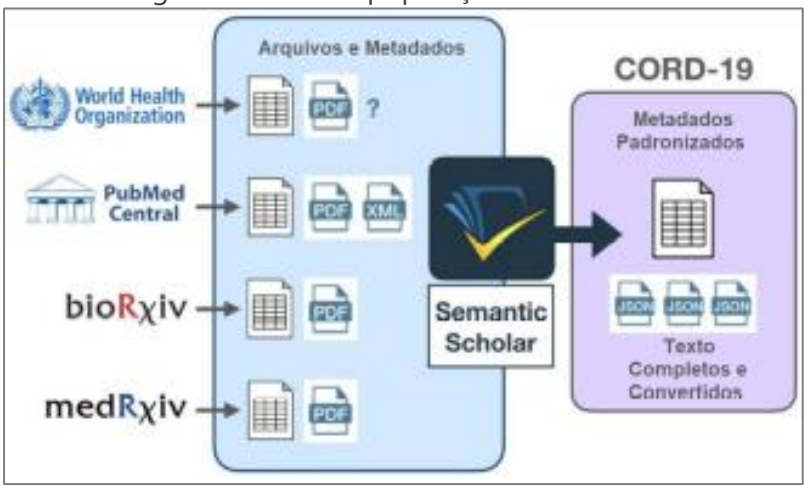
O desenvolvimento da aplicação foi realizado utilizando a linguagem de programação Python, em sua versão 3.9.7, em conjunto com a IDE PyCharm, versão da comunidade 2021.2 RC, sobre o sistema operacional Manjaro Linux, em sua versão 21.1.6. Em procedimentos relacionados ao processamento de linguagem natural, foi utilizada a biblioteca de conjunto de ferramentas de linguagem natural (https://www.nltk.org/), do inglês *Natural Language Toolkit* (NLTK), que possui uma vasta documentação de suas interfaces de programação de aplicativos, do inglês *Application Programming Interface* (API), dos fundamentos computacionais e linguísticos, sendo muito utilizada em estudos, pesquisas e na indústria.

3.1 Aquisição dos dados

Os artigos científicos foram recuperados do repositório CORD-19, que é um agregador de trabalhos de várias fontes como PubMed, Organização das Nações Unidas, bioRxiv e medRxiv. Todos os trabalhos nesse repositório contêm, em um arquivo *Comma Separated Values* (CSV), ao menos o título e seus metadados, os que são de livre acesso e contêm seus textos convertidos em *JavaScript Object Notation* (JSON), figura 3 (Wang *et al.*, 2020).

Foram recuperados, utilizando o *link* de *download* do CORD-19 (https://www.kaggle.com/allen-institute-for-ai/CORD-19-research-challenge), os dados atualizados até o dia 30 de agosto de 2021, quando os metadados possuíam 768.929 registros.

Figura 3: Fluxo de população do CORD-19



Fonte: Adaptado de Wang *et al.* (2020).

3.2 Filtragem dos dados

Utilizando os metadados disponibilizados pelo CORD-19, foram aplicadas técnicas para filtrar os dados. Essa etapa foi dividida em quatro subetapas: filtragem por data, filtragem por palavras, filtragem por linguagem de escrita e remoção de registros duplicados.

Na primeira subetapa, para cada registro do arquivo CSV, foi verificada a data de publicação, sendo considerados apenas os que foram publicados entre 1º de novembro de 2019 e 30 de agosto de 2021 e os que possuíam apenas o ano da publicação. Os que não possuíam esse dado foram desconsiderados.

Com o intuito de selecionar os registros referentes ao Brasil e explicitamente sobre a Covid-19, na segunda subetapa, foi verificado se os títulos e os resumos dos artigos possuíam os termos "brazil", "sao paulo" ou "rio de janeiro" e "covid-19", "coronavirus" ou "sars-cov-2". Durante as validações, os textos foram convertidos para o formato de letra minúscula e os acentos e caracteres especiais foram removidos.

Ao analisar os dados filtrados até a subetapa anterior, observou-se que, em sua maioria, os artigos foram escritos em inglês. Portanto, para que houvesse consistência nas etapas que sucedem a filtragem, a terceira subetapa objetivou remover artigos que não foram escritos em inglês.

Por receber trabalhos de diversas fontes, o CORD-19 pode ter diversos registros sobre o mesmo artigo, portanto, com o intuito de evitar dados duplicados, foi necessário manter apenas um dos registros. Portanto, na quarta subetapa, por meio de procedimentos manuais e automáticos, foram removidos dados duplicados.

As filtragens resultaram em 1.038 artigos, os quais foram pré-processados em seguida.

3.3 Pré-processamento

Cada artigo, da introdução à conclusão, passou por diversos procedimentos de pré-processamento de dados. Primeiramente, foram removidas sentenças não relevantes para a pesquisa, como informações sobre o contexto em que ele foi publicado e sobre licenças de publicação. Após, utilizando a biblioteca NLTK, eles passaram pela etapa de tokenização de sentenças, que consistiu em separar o texto pelas suas sentenças, e de palavras, em que as sentenças foram separadas pelas suas palavras. Em seguida os artigos foram submetidos ao algoritmo de Viterbi, o qual foi treinado com 80% do corpus Treebank, disponibilizado pela ferramenta NLTK, para marcação gramatical das palavras.

Posteriormente foram removidas as *stopwords*, que são palavras irrelevantes para o contexto, e caracteres que não são alfanuméricos. Por fim, foi realizada a lematização das palavras, deflexão para suas formas base, a partir da sua classificação gramatical. Após, os artigos foram agrupados por tópicos e os mais relevantes para a pesquisa foram selecionados.

3.4 Agrupamento

A fim de realizar o agrupamento, com base nos 1.038 artigos, foi calculado o tamanho da amostra, considerando um grau de confiança de 95% e uma margem de erro de 5%. O cálculo resultou em 281 registros, os quais foram selecionados aleatoriamente.

Utilizando o K-Means, que é um algoritmo que define os agrupamentos pelo centro de massa de suas entidades (Manning; Schûtze, 1999), sobre a amostragem, foram criados diversos modelos variando o número de *K*, que significa a quantidade de agrupamentos, e o número de *seed*, um número arbitrário para geração de conjuntos aleatórios de forma reprodutível. Para validação dos mesmos, foi utilizada a medida *Silhouette*, que calcula a similaridade interna e externa dos agrupamentos, em conjunto com a distância Euclidiana (Bonaccorso, 2020). O modelo que apresentou a melhor pontuação, de 0,02719, possuía 6 como número de *K* e 7 como número de *seed*. Os grupos gerados pelo modelo estão apresentados no quadro 2.

Quadro 2: Grupos gerados pelo algoritmo de agrupamento K-Means, com *K* 6 e *seed* 7

Grupo	***20 palavras mais importantes***
1	*social; participant; anxiety; physical; covid; stress; health; study; mental; pandemic; student; depression; psychological; woman; activity; religious; food; score; sleep; exercise*
2	*patient; covid; study; hospital; clinical; mortality; case; sarscov; disease; day; age; symptom; death; hospitalization; group; icu; admission; treatment; care; risk*
3	*variant; mutation; lineage; genome; sequence; sarscov; rbd; spike; antibody; protein; viral; clade; binding; residue; sample; using; sequencing; ace; ek; assay*
4	*model; case; number; covid; parameter; rate; day; epidemic; data; infected; value;*

	equation; curve; individual; time; country; growth; fig; population; reproduction
5	*covid; case; health; state; death; pandemic; data; brazil; city; country; social; region; government; municipality; cluster; number; index; disease; measure; air*
6	*vaccine; vaccination; covid; study; test; sarscov; sample; infection; prevalence; coronavac; antibody; individual; group; age; estimate; ci; seroprevalence; population; death; assay*

Fonte: Dos autores.

Dos grupos gerados, apenas o grupo 1 apresentou termos relevantes referentes as condutas medicamentosas e prognósticos, portanto apenas ele foi escolhido. Após selecioná-lo, todos os artigos foram submetidos ao modelo de agrupamento a fim de classificá-los e selecionar apenas os que estivessem no grupo 1, resultando em 229 trabalhos. Em seguida, eles foram sumarizados.

3.5 Sumarização

Com o intuito de realizar a sumarização extrativa, para cada texto foram criadas as matrizes TF, de frequência das palavras, e IDF, de raridade das palavras, gerando, por fim, a matriz TF-IDF. Cada sentença teve sua pontuação definida pela soma das pontuações das palavras presentes nela, sendo as pontuações recuperadas da matriz TF-IDF. Palavras que não estavam presentes nessa matriz foram consideradas com pontuação Zero. As sentenças que possuíam maior pontuação foram usadas para construir o resumo.

3.6 Experimentos

Os textos selecionados foram submetidos ao modelo da pesquisa e comparados com o modelo TextRank. TextRank é um modelo de classificação baseada em grafos que, recursivamente, decide a importância de um vértice, podendo este ser uma palavra, uma sentença ou outros dados, baseando-se na informação obtida do grafo por completo. Quando um vértice se liga a outro, é registrada uma pontuação para o outro vértice. Quanto mais vértices se ligarem a ele, maior será sua pontuação. Ao final, os que possuírem maiores pontuações são selecionados (Mihalcea; Tarau, 2004). Para aplicar o modelo TextRank, foi utilizada a ferramenta SUMMA, que possui a implementação do modelo.

Sumários eficientes devem apresentar entre 15% e 35% do texto original (Mitkov, 2004), portanto foram realizados experimentos com percentuais dentro dessa faixa, sendo os selecionados: 15%, 25% e 35%. Eles, então, foram validados com o conjunto de métricas ROUGE.

A fim de indicar a significância estatística dos resultados dos experimentos, foram realizadas análises inferenciais com nível de significância $\alpha = 0,05$, ou seja, 95% de confiança, por meio do pacote estatístico para ciências sociais da IBM, versão 25.0. A investigação da distribuição das variáveis, quanto à normalidade, foi realizada por meio da aplicação do teste Kolmogorov-Smirnov. As métricas que apresentaram os dois conjuntos normais foram

avaliadas pelo teste de Levene seguido dos testes t de Student. Contudo, as que apresentaram ao menos um conjunto não normal foram avaliadas por meio do teste U de Mann-Whitney.

4. Resultados e discussões

Dados acerca de condutas medicamentosas e prognósticos puderam ser observados nos resumos gerados, alguns exemplos deles são: a pesquisa realizada por Fonseca *et al.* (2020), sugere que o uso de ivermectina, azitromicina e oseltamivir, além da hidroxicloroquina e prednisona, não provoca redução significativa nos números de hospitalização; o estudo de Emani *et al.* (2020), indica que os testes de hidroxicloroquina, realizado pelo Brasil, e de hidroxicloroquina e lopinavir/ritonavir, realizado pela Organização Mundial da Saúde, não mostraram benefícios para redução da mortalidade; o trabalho de Lamback *et al.* (2021), que cita alguns estudos que sugerem que a aplicação de azitromicina com hidroxicloroquina reduziu a carga viral do Covid-19; e a pesquisa de Borba *et al.* (2020), a qual indica que o uso da cloroquina ocasionou alguns problemas de saúde, como miopatia e rabdomiólise.

Com base nos trabalhos correlatos, para validar os resultados foram utilizados os conjuntos de métricas ROUGE-1, ROUGE-2 e ROUGE-L. ROUGE para avaliação de sumarização automática, a qual é comparada com uma ou mais referências criadas por humanos. ROUGE-N refere-se à sobreposição de n-gramas do sumário produzido na referência, sendo N o tamanho dos n-gramas, por exemplo: ROUGE-1, sobreposição de unigramas; ROUGE-2, sobreposição de bigramas. ROUGE-L retrata a subsequência comum mais longa do sumário produzido sobre a referência. Valores próximos a 0 retratam uma sumarização ruim, enquanto próximos a 1 determinam uma boa sumarização (Lin, 2004).

Foram calculadas as médias das métricas *recall*, precisão e pontuação F1, dos conjuntos ROUGE-1, ROUGE-2 e ROUGE-L, para as sumarizações com 15%, 25% e 35% do texto original, realizadas pelo modelo proposto e pelo modelo TextRank.

Recall, demonstrada na equação 3, refere-se à proporção de predições corretas que o sistema fez, sobre as predições corretas e o que deveria ser predito.

$$recall = \frac{\text{verdadeiros positivos}}{\text{verdadeiros positivos} + \text{falsos negativos}} \quad (3)$$

A precisão, apresentada na equação 4, é a proporção de predições corretas que o sistema fez, sobre todas as predições.

$$\text{precisão} = \frac{\text{verdadeiros positivos}}{\text{verdadeiros positivos} + \text{falsos positivos}} \quad (4)$$

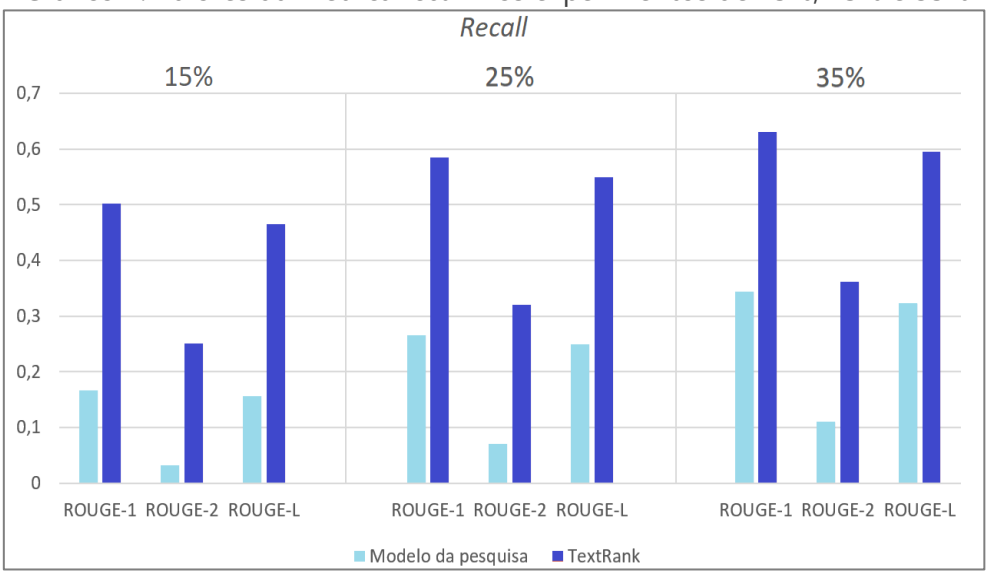
O cálculo realizado entre *recall* e precisão, retratado pela equação 5, gera a pontuação F1.

(5)

$$F1 = \frac{2 \text{ x precisão x } \textit{recall}}{\text{precisão} + \textit{recall}}$$

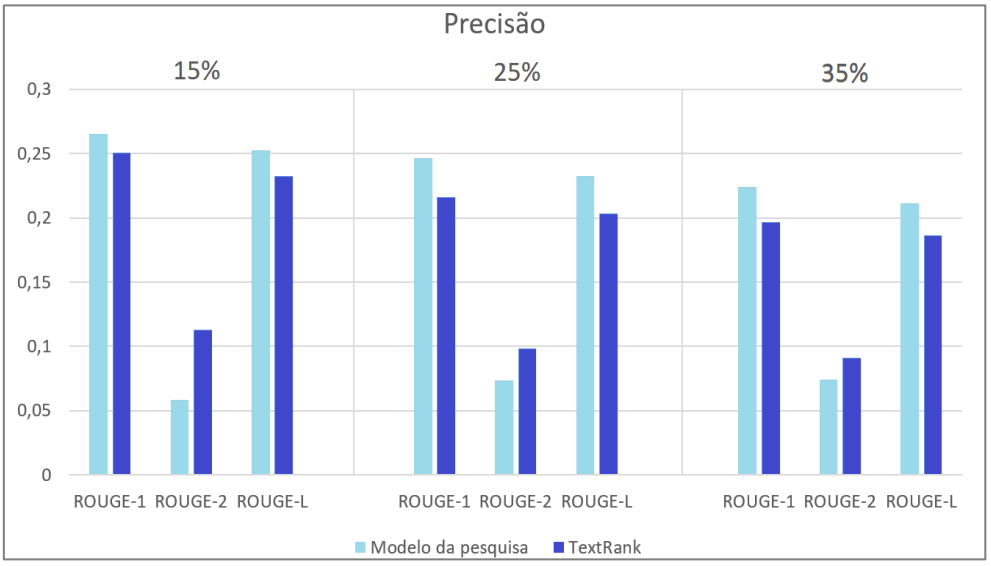
Por meio dos resultados dos experimentos, demonstrados nos gráfico 1, gráfico 2 e gráfico 3, é possível observar que em todas as métricas o modelo da pesquisa apresentou valores de *recall* inferiores ao do modelo TextRank. Em relação a precisão, ele foi superior nas métricas ROUGE-1 e ROUGE-L, enquanto na métrica ROUGE-2 o modelo TextRank teve melhores resultados. Em decorrência dos baixos valores de *recall*, em todas as métricas o modelo da pesquisa também apresentou valores inferiores ao modelo TextRank, para a pontuação F1.

Gráfico 1: Valores da métrica *recall* nos experimentos de 15%, 25% e 35%



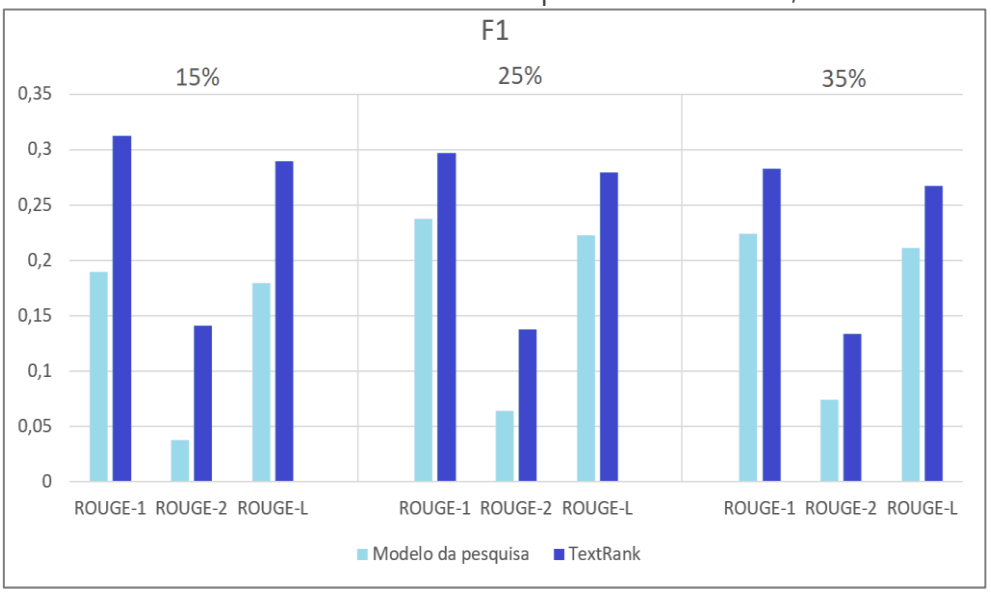
Fonte: Dados da pesquisa (2024).

Gráfico 2: Valores da métrica precisão nos experimentos de 15%, 25% e 35%



Fonte: Dados da pesquisa (2024).

Gráfico 3: Valores da métrica F1 nos experimentos de 15%, 25% e 35%



Fonte: Dados da pesquisa (2024).

Os valores para média, desvio padrão e p podem ser vistos nas tabela 1, tabela 2 e tabela 3. Diante dos resultados estatísticos obtidos, é inferido que todas as comparações de métricas tiveram significância estatística, pois possuem valores p menores ou iguais a 0,05. Portanto, em relação à precisão nos conjuntos ROUGE-1 e ROUGE-L, o modelo da pesquisa, que utilizou o algoritmo de Viterbi e TF-IDF, teve melhor desempenho. Contudo, nos demais resultados, o modelo da pesquisa apresentou valores menores em comparação com os do TextRank, significando um desempenho inferior.

Tabela 1: Valores estatísticos para a métrica *recall* nos experimentos de 15%, 25% e 35%

Experimento	Métrica	Modelo da pesquisa Média ± DP	TextRank Média ± DP	Valor-p
15%	ROUGE-1	0,17 ± 0,07	0,50 ± 0,13	< 0,001*
	ROUGE-2	0,03 ± 0,03	0,26 ± 0,12	< 0,001**
	ROUGE-L	0,16 ± 0,07	0,46 ± 0,12	< 0,001*
25%	ROUGE-1	0,27 ± 0,09	0,59 ± 0,14	< 0,001**
	ROUGE-2	0,07 ± 0,05	0,32 ± 0,15	< 0,001**
	ROUGE-L	0,25 ± 0,09	0,55 ± 0,14	< 0,001**
35%	ROUGE-1	0,34 ± 0,11	0,63 ± 0,14	< 0,001**
	ROUGE-2	0,11 ± 0,07	0,36 ± 0,16	< 0,001**
	ROUGE-L	0,32 ± 0,10	0,60 ± 0,14	< 0,001*

*Valor obtido após aplicação do teste t de Student para amostras independentes.
**Valor obtido após aplicação do teste U de Mann-Whitney.

Fonte: Dados da pesquisa (2024).

Tabela 2: Valores estatísticos para a métrica precisão nos experimentos de 15%, 25% e 35%

Experimento	Métrica	Modelo da pesquisa Média ± DP	TextRank Média ± DP	Valor-p
15%	ROUGE-1	0,27 ± 0,10	0,25 ± 0,12	0,016**
	ROUGE-2	0,06 ± 0,07	0,11 ± 0,10	< 0,001**
	ROUGE-L	0,25 ± 0,10	0,23 ± 0,11	0,003**
25%	ROUGE-1	0,25 ± 0,10	0,22 ± 0,10	< 0,001**
	ROUGE-2	0,07 ± 0,09	0,10 ± 0,09	< 0,001**
	ROUGE-L	0,23 ± 0,10	0,20 ± 0,10	< 0,001**
35%	ROUGE-1	0,22 ± 0,10	0,20 ± 0,10	< 0,001**
	ROUGE-2	0,07 ± 0,09	0,09 ± 0,09	< 0,001**
	ROUGE-L	0,21 ± 0,10	0,19 ± 0,10	< 0,001**

*Valor obtido após aplicação do teste t de Student para amostras independentes.
**Valor obtido após aplicação do teste U de Mann-Whitney.

Fonte: Dados da pesquisa (2024).

Tabela 3: Valores estatísticos para a métrica F1 nos experimentos de 15%, 25% e 35%

Experimento	Métrica	Modelo da pesquisa Média ± DP	TextRank Média ± DP	Valor-p
15%	ROUGE-1	0,19 ± 0,06	0,31 ± 0,09	< 0,001*
	ROUGE-2	0,04 ± 0,03	0,14 ± 0,08	< 0,001**
	ROUGE-L	0,18 ± 0,06	0,30 ± 0,09	< 0,001*
25%	ROUGE-1	0,24 ± 0,06	0,30 ± 0,09	< 0,001**
	ROUGE-2	0,06 ± 0,05	0,14 ± 0,08	< 0,001**
	ROUGE-L	0,22 ± 0,06	0,28 ± 0,09	< 0,001*
35%	ROUGE-1	0,25 ± 0,06	0,28 ± 0,09	< 0,001**
	ROUGE-2	0,08 ± 0,05	0,13 ± 0,08	< 0,001**
	ROUGE-L	0,24 ± 0,06	0,27 ± 0,09	< 0,001**

*Valor obtido após aplicação do teste t de Student para amostras independentes.
**Valor obtido após aplicação do teste U de Mann-Whitney.

Fonte: Dados da pesquisa (2024).

A partir dos resultados, deduz-se que, embora o algoritmo de Viterbi auxilie na classificação gramatical das palavras, passo importante para lematização e, consequentemente, para definição de importância delas, a aplicação isolada de TF-IDF não tem performance satisfatória para extração de informações relevantes de textos. Isso vem ao encontro dos resultados demonstrados pela pesquisa de Zaware *et al.* (2021), na qual a aplicação isolada de TF-IDF se mostrou inferior à aplicação em conjunto com TextRank. A dedução também corrobora com os achados de Rani e Lobiyal (2021), os quais demonstram que o TF-IDF pode apresentar bom desempenho para sumarização de textos, desde que outras abordagens sejam utilizadas em conjunto.

A utilização de TF-IDF com *word banks*, para sumarização de artigos científicos do CORD-19, empregada na pesquisa de Jain, Bellaney e Jangid (2021), apresentou desempenho superior ao de ferramentas online, reforçando o pressuposto de que TF-IDF em conjunto com outras abordagens gera melhores resultados. Em contraponto, Rai *et al.* (2021) demonstraram que a sumarização guiada de artigos científicos do CORD-19, por meio de BERT e outras abordagens, gera melhores resultados, quando comparados aos do TextRank.

5. Considerações finais

Nesta pesquisa foi aplicado o algoritmo de Viterbi em conjunto com TF-IDF para sumarização de textos médico-científicos relacionados à Covid-19 no Brasil. Os dados foram obtidos do repositório CORD-19, filtrados, pré-processados e, com o algoritmo K-means, agrupados por tópicos, sendo selecionados apenas os relevantes para a pesquisa. Para avaliação das sumarizações, foram utilizadas as métricas ROUGE-1, ROUGE-2 e ROUGE-L, em comparação com o método TextRank.

Os resultados da abordagem aqui proposta, em suma, mostraram desempenho inferior em relação ao modelo TextRank. Apesar de resultados satisfatórios referentes à precisão, os valores de *recall* e F1 se apresentaram contrários ao esperado.

Em trabalhos futuros na área, sugere-se a realização de pesquisas utilizando o classificador Naive Bayes em conjunto com HMM e com o conjunto de dados do Sistema Unificado de Linguagem Médica, que oferece diversas informações, classificações e terminologias biomédicas. Ademais, sugere-se a validação do modelo em bases de dados homogêneas e já conceituadas, com textos próprios para referências e análises, como o DUC.

Referências bibliográficas

APPOLINÁRIO, F. **Metodologia da ciência**: filosofia e prática da pesquisa**.** 2. ed. São Paulo: Cengage Learning, 2012.

BONACCORSO, Giuseppe. **Machine Learning Algorithms**: a reference guide to popular algorithms for data science and machine learning. Birmingham: Packt, 2017.

GOULARTE, F. *et al.* A text summarization method based on fuzzy rules and applicable to automated assessment. **Expert Systems with Applications**, v. 115, p. 264-275, jan. 2019. DOI: https://doi.org/10.1016/j.eswa.2018.07.047.

HIRSCHBERG, J.; MANNING, C. D. Advances in natural language processing. **Science**, v. 349, n. 6245, 17 jul. 2015. DOI: 10.1126/science.aaa8685.

HOSNY, A.; AERTS, H. J. W. L. Artificial intelligence for global health. **Science**, v. 366, n. 6468, 22 nov. 2019. DOI: 10.1126/science.aay5189.

INDURKHYA, N.; DAMERAU, F. J. **Handbook of Natural Language Processing.** Chapman & Hall/CRC, 2010.

LIN, C. ROUGE: A Package for Automatic Evaluation of Summaries. **Text Summarization Branches Out**, 2004.

LOPES, R. J. Pandemia de coronavírus gera corrida global por vacinas, medicamentos e diagnósticos. Folha de São Paulo, 2020. Disponível em: <https://www1.folha.uol.com.br/equilibrioesaude/2020/03/pandemia-de-coronavirus-gera-corrida-global-por-vacinas-medicamentos-e-diagnosticos.shtml>. Acesso em: 18 mar. 2024.

MANNING, C. D.; SCHÛTZE, H. **Foundations of Statistical Natural Language Processing.** Cambridge: The MIT Press, 1999.

MIHALCEA, R.; TARAU, P. TextRank: Bringing Order into Text. Proceedings of the 2004 **Conference on Empirical Methods in Natural Language Processing**, 2004.

MITKOV, Ruslan. **The Oxford Handbook of Computational Linguistics.** Oxônia: Oup Oxford, 2004.

MORADI, M.; DORFFNER, G.; SAMWALD, M. Deep contextualized embeddings for quantifying the informative content in biomedical text summarization. **Computer Methods and Programs in Biomedicine**, v. 184, fev. 2020. DOI: https://doi-org/10.1016/j.cmpb.2019.105117.

NEILL, D. B. New Directions in Artificial Intelligence for Public Health Surveillance. **IEEE Intelligent Systems**, v. 27, n. 1, jan. 2012. DOI: 10.1109/MIS.2012.18.

OLIVEIRA, H. T. A. **Sumarização Automática de Textos Baseada em Conceitos via Programação Linear Inteira e Regressão.** Universidade Federal de Pernambuco, Recife, 2018.

PHAM, Q.-V. *et al.* Artificial Intelligence (AI) and Big Data for Coronavirus (COVID-19) Pandemic: A Survey on the State-of-the-Arts. **IEEE Access**, v. 8, 2020. DOI: 10.1109/ACCESS.2020.3009328.

RANI, R.; LOBIYAL, D. K. A weighted word embedding based approach for extractive text summarization. **Expert Systems with Applications**, v. 186, dez. 2021. DOI: 10.1016/j.eswa.2021.115867.

RUSSEL, S.; NORVIG, P. **Inteligência Artificial.** 3. ed. Rio de Janeiro: Elsevier Editora Ltda, 2013.

VAJJALA, S. *et al.* **Practical Natural Language Processing.** O'Reilly Media, Inc., 2020.

WANG, L. L. *et al.* **CORD-19: The COVID-19 Open Research Dataset.** 22 abr. 2020.

ZAWARE, Sarika *et al*. Text Summarization using TF-IDF and Textrank algorithm. 2021. *In:* 5TH INTERNATIONAL CONFERENCE ON TRENDS IN ELECTRONICS AND INFORMATICS (ICOEI), p. 1399-1407, jun. 2021. DOI: 10.1109/ICOEI51242.2021.9453071.