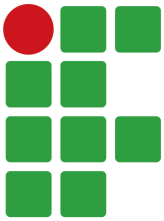


INSTITUTO FEDERAL
Goiás
Câmpus Anápolis

Paulo David Silva Dias

Comparação entre Abordagem Neurais e Não Neurais para Modelagem de Tópicos em Tweets sobre Institutos Federais Brasileiros.

Anápolis-GO

2025

Paulo David Silva Dias

Comparação entre Abordagem Neurais e Não Neurais para Modelagem de Tópicos em Tweets sobre Institutos Federais Brasileiros.

Projeto de Pesquisa apresentado ao curso de Bacharelado em Ciência da Computação, como requisito para obtenção do grau final na disciplina de Projeto Final de Curso.

Instituto Federal de Goiás - Câmpus Anápolis

Orientador: Prof. Dr. Sérgio Daniel Carvalho Canuto

Anápolis-GO
2025

Dados Internacionais de Catalogação na Publicação (CIP)

Dias, Paulo David Silva.

D541c Comparação entre abordagem neurais e não neurais para modelagem de tópicos em tweets sobre Institutos Federais brasileiros. / Paulo David Silva Dias. – Anápolis: IFG, 2025.
90 f. il. color.

Orientador: Prof. Dr. Sérgio Daniel Carvalho Canuto.

Trabalho de Conclusão de Curso – Instituto Federal de Educação, Ciência e Tecnologia de Goiás: Campus Anápolis – Bacharelado em Ciência da Computação, 2025.

1. Redes neurais. 2. modelagem de tópicos. 3. tweets. 4. Institutos Federais.
I. Canuto, Sérgio Daniel Carvalho (orient.).
II. Título

CDD 004.5

Ficha catalográfica elaborada pela bibliotecária Claudineia Pereira de Abreu CRB1/1956
IFG Campus Anápolis

TERMO DE AUTORIZAÇÃO PARA DISPONIBILIZAÇÃO NO REPOSITÓRIO DIGITAL DO IFG - ReDi IFG

Com base no disposto na Lei Federal nº 9.610/98, AUTORIZO o Instituto Federal de Educação, Ciência e Tecnologia de Goiás, a disponibilizar gratuitamente o documento no Repositório Digital (ReDi IFG), sem ressarcimento de direitos autorais, conforme permissão assinada abaixo, em formato digital para fins de leitura, download e impressão, a título de divulgação da produção técnico-científica no IFG.

Identificação da Produção Técnico-Científica

[ ] Tese	[ ] Artigo Científico
[ ] Dissertação	[ ] Capítulo de Livro
[ ] Monografia – Especialização	[ ] Livro
[X] TCC - Graduação	[ ] Trabalho Apresentado em Evento
[ ] Produto Técnico e Educacional - Tipo: _____	

Nome Completo do Autor: Paulo David Silva Dias
Matrícula: 20181060140226
Título do Trabalho: Comparação entre Abordagem Neurais e Não Neurais para Modelagem de Tópicos em Tweets sobre Institutos Federais Brasileiros.

Autorização - Marque uma das opções

1. (X) Autorizo disponibilizar meu trabalho no Repositório Digital do IFG (acesso aberto);
2. ( ) Autorizo disponibilizar meu trabalho no Repositório Digital do IFG somente após a data ___/___/_____ (Embargo);
3. ( ) Não autorizo disponibilizar meu trabalho no Repositório Digital do IFG (acesso restrito).

Ao indicar a opção **2 ou 3**, marque a justificativa:

( ) O documento está sujeito a registro de patente.
( ) O documento pode vir a ser publicado como livro, capítulo de livro ou artigo.
( ) Outra justificativa: _____

DECLARAÇÃO DE DISTRIBUIÇÃO NÃO-EXCLUSIVA

O/A referido/a autor/a declara que:

i. o documento é seu trabalho original, detém os direitos autorais da produção técnico-científica e não infringe os direitos de qualquer outra pessoa ou entidade;
ii. obteve autorização de quaisquer materiais inclusos no documento do qual não detém os direitos de autor/a, para conceder ao Instituto Federal de Educação, Ciência e Tecnologia de Goiás os direitos requeridos e que este material cujos direitos autorais são de terceiros, estão claramente identificados e reconhecidos no texto ou conteúdo do documento entregue;
iii. cumpriu quaisquer obrigações exigidas por contrato ou acordo, caso o documento entregue seja baseado em trabalho financiado ou apoiado por outra instituição que não o Instituto Federal de Educação, Ciência e Tecnologia de Goiás.

gov.br Documento assinado digitalmente
PAULO DAVID SILVA DIAS
Data: 22/09/2025 10:32:39-0300
Verifique em https://validar.iti.gov.br

Anápolis-GO, Local — 27/03/2025. Data

_____
Assinatura do Autor e/ou Detentor dos Direitos Autorais

ATA DA SESSÃO PÚBLICA DE APRESENTAÇÃO E DEFESA DO TRABALHO DE CONCLUSÃO DE CURSO

Ao 27 dias do mês de Março do ano de dois mil e vinte e cinco, às 15:30 horas, foi realizada a sessão pública de apresentação e defesa do Trabalho de Conclusão de Curso do Graduando Paulo David Silva Dias (matrícula 20181060140226) do curso de Bacharelado em Ciência da Computação, no primeiro semestre do ano de dois mil e dezoito. A banca foi composta pelos seguintes membros: Dr. Sérgio Daniel Carvalho Canuto, Me. Alexandre Bellezi José e Dr. Geraldo Witeze Junior sob a presidência do primeiro. O trabalho de conclusão de curso tem como título **"Comparação entre Abordagem Neurais e Não Neurais para Modelagem de Tópicos em Tweets sobre Institutos Federais Brasileiros"**, da área de Ciência da Computação, sob orientação do Prof. Sérgio Daniel Carvalho Canuto. Após a apresentação do Trabalho de Conclusão de Curso, tendo sido o autor arguido pela Banca Examinadora, a nota obtida foi 9,0.

Encerra-se a presente sessão às 17 horas e 30 minutos. Eu, Prof.Sérgio Daniel Carvalho Canuto, dato e assino a presente ata que segue assinada por todos os membros da Banca e pelo graduando.

Documento assinado eletronicamente por:

- **Paulo David Silva Dias**, **20181060140226 - Discente**, em 04/04/2025 08:35:26.
- **Geraldo Witeze Junior**, **PROFESSOR ENS BASICO TECN TECNOLOGICO**, em 03/04/2025 22:37:33.
- **Alexandre Bellezi Jose**, **PROFESSOR ENS BASICO TECN TECNOLOGICO**, em 03/04/2025 08:50:52.
- **Sergio Daniel Carvalho Canuto**, **PROFESSOR ENS BASICO TECN TECNOLOGICO**, em 28/03/2025 15:14:05.

Este documento foi emitido pelo SUAP em 28/03/2025. Para comprovar sua autenticidade, faça a leitura do QRCode ao lado ou acesse https://suap.ifg.edu.br/autenticar-documento/ e forneça os dados abaixo:

Código Verificador: 633002
Código de Autenticação: 0bf9a413a6

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

MT	Modelagem de Tópicos
PLN	Processamento de Linguagem Natural
IFs	Institutos Federais Brasileiros
LDA	Latent Dirichlet Allocation
NMF	Non-Negative Matrix Factorization
BERTopic	Topic Modeling Based on BERT Representations
RNA	Redes Neurais Artificiais
MLP	Multilayer Perceptron
ReLU	Rectified Linear Units
KNN	Knowledge Discovery in Databases
GloVe	Global Vectors for Word Representation
ELMo	Embeddings from Language Models
GPT	Generative Pre-trained Transformer
BERT	Bidirectional Encoder Representations from Transformers
MSE	Mean Squared Error
PE	Positional Encoding
WE	Word Embedding
MLM	Masked Language Modeling
NSP	Next Sentence Prediction
MASK	Máscara
CLS	Classification Token
SEP	Separator Token
SBERT	Sentence BERT

UMAP	Uniform Manifold Approximation and Projection
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
HDBSCAN	Hierarchical Density-Based Spatial Clustering of Applications with Noise
TF-IDF	Term Frequency-Inverse Document Frequency
c-TF-IDF	Class-based Term Frequency-Inverse Document Frequency
FNN	Feedforward Neural Network
CV	Cross Validation
PMI	Ponto de informação mútua
RBO	Rank Biased Overlap

Lista de símbolos

α	Alpha
β	Beta
θ	Theta
η	Eta
$\int$	Integral
$\prod$	Produto
$\sum$	Soma
σ	Sigma
ϵ	Epsilon
∂	Derivada Parcial
$\surd$	Raiz Quadrada
$<$	Menor
$\leq$	Menor ou igual

Sumário

1 Introdução

A análise de comentários em redes sociais desempenha hoje uma função essencial na compreensão das percepções da comunidade, especialmente no âmbito das instituições públicas, como os Institutos Federais Brasileiros (IFs). Como enfatizado por Alamsyah et al. (2018), os fluxos digitais exercem uma influência global significativa, promovendo conectividade e divulgação de informações. A rápida análise de dados é imprescindível, sobretudo nas plataformas de mídia social, onde é possível extrair e analisar opiniões de seus usuários.

Esta análise desempenha um papel crucial na monitoração em tempo real do sentimento geral de usuários sobre variados tópicos, na previsão de crises e na avaliação do apoio ou desaprovação das medidas institucionais. Além disso, como destacado por Jr, Tavares e Kido (2017), independente do contexto da análise textual, a caracterização do conteúdo do texto é uma tarefa essencial. Isso pode ser realizado por meio da identificação de categorias ou palavras-chave que representam o tema central. Existem várias abordagens para identificar tais palavras-chave, sendo as soluções de Modelagem de Tópicos (MT) de maior destaque dentre as abordagens para esse problema de caracterização de conteúdo.

Nas mídias sociais, onde há um excesso de informações sendo geradas constantemente, a Modelagem de Tópicos tem o potencial de auxiliar na automação da extração e visualização de informações relevantes sobre como as pessoas percebem e interagem com as instituições públicas. Ao identificar os principais tópicos de discussão, é possível elencar potenciais preocupações, interesses e sentimentos do público em relação a essas instituições, bem como monitorar a reputação de instituições, identificando potenciais crises de imagem ou questões emergentes que demandam atenção.

A Modelagem de Tópicos é amplamente empregada em contextos como saúde (PINTO et al., 2020) e política (CARMO et al., 2023; HOANG; COHEN; LIM, 2014). No contexto de análise de comentários sobre instituições de ensino do presente trabalho, estudos anteriores aplicaram estratégias de Modelagem de Tópicos na compreensão das experiências dos estudantes (HONG; PARK; CHOI, 2020), na análise de crises universitárias (SANANDRES; ABELLO; MADARIAGA, 2020), e em sistemas de aprendizado virtual (KUANG; LUO; SUN, 2011). De forma geral, estudos anteriores se concentraram na aplicação de uma única técnica de Modelagem de Tópicos, utilizando predominantemente modelos não neurais tradicionais. Entretanto, a comparação entre diferentes técnicas e métricas de avaliação dos tópicos obtidos em cada contexto é necessária para assegurar a adequação de técnicas a cada contexto (TERRAGNI et al., 2021).

O presente trabalho se destaca por ser pioneiro ao comparar a efetividade de

abordagens tradicionais com arquiteturas de redes neurais recentes no âmbito da modelagem de tópicos extraídos da rede social Twitter[1] (recentemente renomeada para "X") com recortes inéditos de dados sobre os Institutos Federais Brasileiros (IFs). Neste trabalho são apresentadas análises de modelagem de tópicos com um recorte inicial de dados (com tweets publicados entre agosto de 2023 e agosto de 2024), recortes das 5 macro-regiões do Brasil e recortes de 3 categorias de sentimento (positivos, negativos e neutros). A análise comparativa de técnicas recentes para Modelagem de Tópicos inclui variações do método (baseado em Redes Neurais) BERTopic (Bidirectional Encoder Representations from Transformer) (GROOTENDORST, 2022), bem como métodos não neurais tradicionalmente adotados, como LDA (Latent Dirichlet Allocation) (BLEI; NG; JORDAN, 2003) e NMF (Non-negative Matrix Factorization) (LEE; SEUNG, 1999). O intuito de tais comparações consiste em avaliar qual dessas abordagens é mais adequada para a Modelagem de Tópicos em mensagens curtas (tweets) relacionados aos IFs.

As técnicas de Modelagem de Tópicos baseadas no BERTopic apresentaram resultados na métrica de coerência consistentemente superiores aos obtidos com as técnicas tradicionais (LDA e NMF) em todos os recortes avaliados, visto que o BERTopic é capaz de representar a informação semântica dos termos. Entretanto, ao considerar a diversidade dos tópicos obtidos e a estabilidade das técnicas em diferentes execuções, os métodos tradicionais apresentaram resultados superiores às abordagens baseadas no BERTopic.

Como contribuição final do trabalho, é apresentada a listagem dos tópicos automaticamente identificados com a técnica de melhor coerência. Tal listagem ilustra discussões, preocupações e eventos que emergem do Twitter em cada recorte de dados avaliado. Portanto, é possível constatar que a Modelagem de Tópicos tem o potencial de oferecer a possibilidade de criação de futuras aplicações práticas em tempo real para monitorar potenciais crises, identificar temas emergentes e temas de interesse para comunicação institucional. Em suma, as contribuições do presente trabalho podem ser sumarizadas como:

- A obtenção de tópicos em tweets sobre os IFs, considerando diferentes recortes de sentimentos dos tweets e regiões do Brasil.
- Comparação e análise da efetividade de técnicas avançadas de Modelagem de Tópicos no contexto de tweets sobre IFs considerando métricas de coerência, diversidade, estabilidade e tempo de execução dos métodos.
- Apresentação de ampla fundamentação teórica sobre técnicas de modelagem neurais e não neurais, bem como trabalhos que envolvem Modelagem de Tópicos no contexto educacional e no contexto de redes sociais.

[1] A nomenclatura Twitter foi adotada devido à popularidade e não ambiguidade de tal termo. Twitter foi a nomenclatura oficial durante o período da coleta dos dados utilizados neste trabalho.

O restante deste trabalho está organizado da seguinte forma: Na Seção 2, apresenta-se o referencial teórico em redes neurais. Na Seção 3, discute-se o referencial teórico relacionado à modelagem de tópicos. Na Seção 4, descreve-se a metodologia adotada. Na Seção 5, são apresentados os resultados experimentais. Por fim, na Seção 6, são discutidas as conclusões do trabalho.

2 Referencial Teórico sobre Redes Neurais

Esta seção visa uma apresentação detalhada dos componentes necessários para a construção da técnica de Modelagem Tópicos no estado da arte, denominada BERTopic (Bidirectional Encoder Representations from Transformers)(GROOTENDORST, 2022). Portanto, serão revisados conceitos sobre o aprendizado baseado em redes neurais, onde serão discutidos seus princípios fundamentais e aplicações. A exploração deste tema é organizada em quatro seções distintas: base conceitual das redes neurais, fundamentos de representações de palavras com redes neurais artificiais, fundamentos da arquitetura *Transformers* e o BERT.

2.1 Base Conceitual das Redes Neurais

Um neurônio biológico, conforme abordado por (GÉRON, 2017), é encontrado no córtex cerebral, sendo composto por:

- Corpo celular: Contém o núcleo e a maioria dos componentes complexos da célula, juntamente com diversas extensões ramificadas.
- Dendritos: Os dendritos são responsáveis por captar sinais de outros neurônios por meio de estruturas sinápticas.
- Axônio: uma extensão muito longa do neurônio, é responsável por transmitir sinais elétricos para outras células.

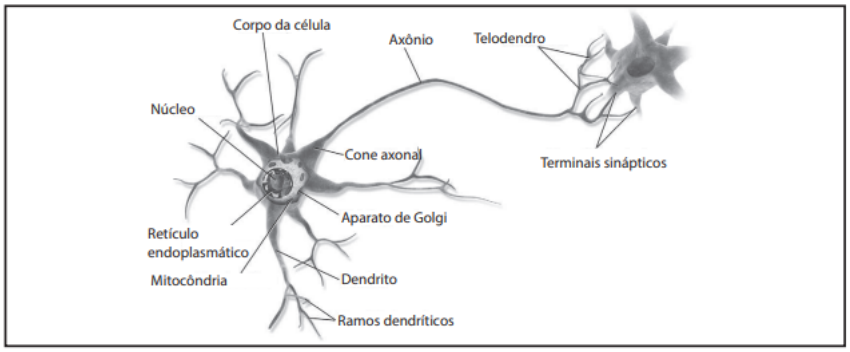


Figura 1 – Neurônio biológico. Fonte: (GÉRON, 2017)

Conforme apresentado na Figura 1, próximo à sua extremidade, o axônio divide-se em vários ramos chamados de telodendritos, e na ponta desses ramos estão localizadas as estruturas minúsculas chamadas terminais sinápticos, que estão conectadas aos dendritos (ou diretamente ao corpo celular) de outros neurônios. Os neurônios biológicos estão

organizados em uma vasta rede, com cada neurônio geralmente conectado a milhares de outros (GÉRON, 2017).

Um neurônio artificial é um modelo matemático simplificado inspirado nos neurônios biológicos encontrados no cérebro. Foi proposto por Warren McCulloch e Walter Pitts (MCCULLOCH; PITTS, 1943). Posteriormente, o neurônio artificial Perceptron (ROSENBLATT, 1958), aprimorou a formulação original com inclusão de pesos ajustáveis.

O *Perceptron*, é uma das arquiteturas mais simples de redes neurais artificiais. Em comparação com neurônio biológico, as conexões (axônios), possuem pesos para indicar as forças da conexões, as entradas (terminais sinápticos) são combinadas por meio de uma função de ativação e repassadas para o próximo neurônio.

Uma ilustração do neurônio Peceptron é apresentada na Figura 2.

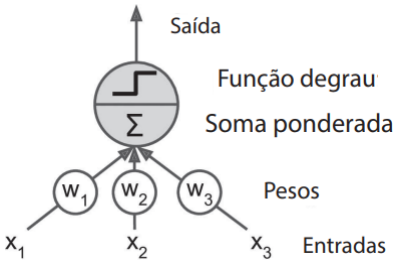


Figura 2 – Exemplo de um Perceptron. Fonte: (GÉRON, 2017)

.

O *Perceptron* realiza uma operação sobre uma ou mais entradas, cada uma associada a um peso. Ele calcula a soma ponderada das entradas, que é representada como:

$$z = w_1 x_1 + w_2 x_2 + \ldots + w_n x_n = \mathbf{w}^T \cdot \mathbf{x}. \tag{2.1}$$

Esse valor é então passado por uma função de ativação, comumente a função degrau, e o resultado é enviado para a saída:

$$h_w(\mathbf{x}) = step(z) = step(\mathbf{w}^T \cdot \mathbf{x}). \tag{2.2}$$

A função degrau mais comum utilizada nos Perceptrons é a função de *Heaviside*:

$$heaviside(z) = \begin{cases} 0 & \text{se } z < 0 \\ 1 & \text{se } z \geq 0 \end{cases} \tag{2.3}$$

Funções de ativação são cruciais nas Redes Neurais Artificiais (RNA), pois determinam a saída de cada perceptron (neurônio). A função de *Heaviside* é uma função de passo que muda de 0 para 1 em $x = 0$. Embora seja simples e útil em muitos casos, ela não é diferenciável em $x = 0$, o que dificulta o treinamento de redes neurais usando o algoritmo de retropropagação (*backpropagation*).

O uso de funções lineares como a função de *Heaviside* pode limitar a RNA, especialmente em problemas não lineares. Para lidar com essa limitação, são necessárias funções não lineares, como a função sigmoide.

Por outro lado, a função sigmoide, também conhecida como função logística definida por:

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{2.4}$$

É uma função de ativação não linear que é diferenciável em todos os lugares. Para viabilizar o uso do algoritmo *backpropagation*, a função degrau foi substituída pela função sigmoide na arquitetura do *Multilayer Perceptron*, também conhecido como MLP. Enquanto a função degrau contém apenas segmentos planos, tornando difícil o ajuste dos parâmetros, a função sigmoide possui uma inclinação definida em todos os pontos, facilitando a otimização da RNA. Além da função sigmoide, o algoritmo de *backpropagation* pode ser utilizado com outras funções de ativação populares, como por exemplo a função tangente hiperbólica e função ReLU (*Rectified Linear Units*) (GÉRON, 2017).

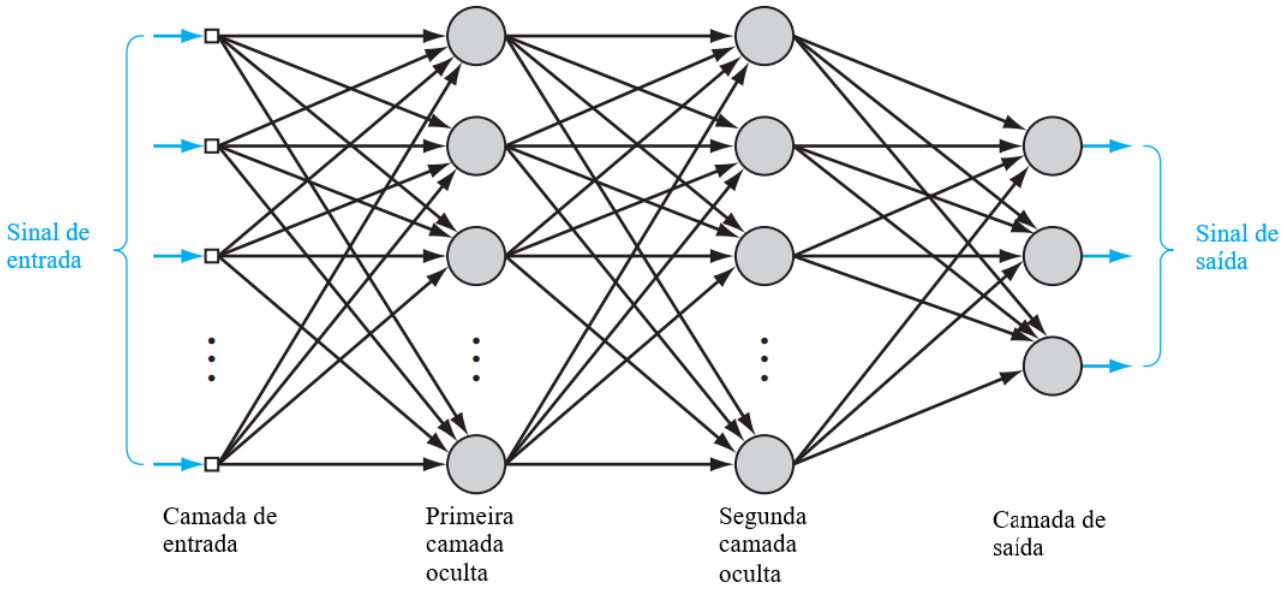


Figura 3 – Exemplo de arquitetura de uma rede neural MLP. Fonte: (HAYKIN, 2009)

.

Um *Multilayer Perceptron* (MLP) é um tipo de rede neural artificial com múltiplos neurônios (*Perceptrons*), onde os neurônios estão interligados entre múltiplas camadas. Ele é composto por uma camada de entrada, responsável por receber a informação que será processada pela rede, uma ou mais camadas ocultas, que são grupos de neurônios entre a camada de entrada e a camada de saída. Quando uma RNA possui duas ou mais camadas ocultas, é chamada de rede neural profunda, também conhecida como *deep learning* (GÉRON, 2017). Na figura 3, o processamento da informação em um MLP ocorre

de forma direta, seguindo um fluxo conhecido como *feedforward*. Nesse fluxo, os dados são propagados da camada de entrada, passando por todas as camadas ocultas, até alcançar a camada de saída, que é responsável por produzir a resposta da rede.

Em 1986, Rumelhart, Hinton e Williams (1986), introduziram o algoritmo de *backpropagation* para lidar com o treinamento do MLP. Este algoritmo foi fundamental para permitir o treinamento eficiente de *deep learning*, viabilizando a aprendizagem de representações complexas dos dados.

Conforme descrito por Géron (2017), o algoritmo de *backpropagation* funciona em duas etapas principais, mostradas na Figura 4. Na primeira etapa, chamada de *forward pass*, cada instância de treinamento é alimentada para a rede, e o algoritmo calcula a saída de cada neurônio em todas as camadas.

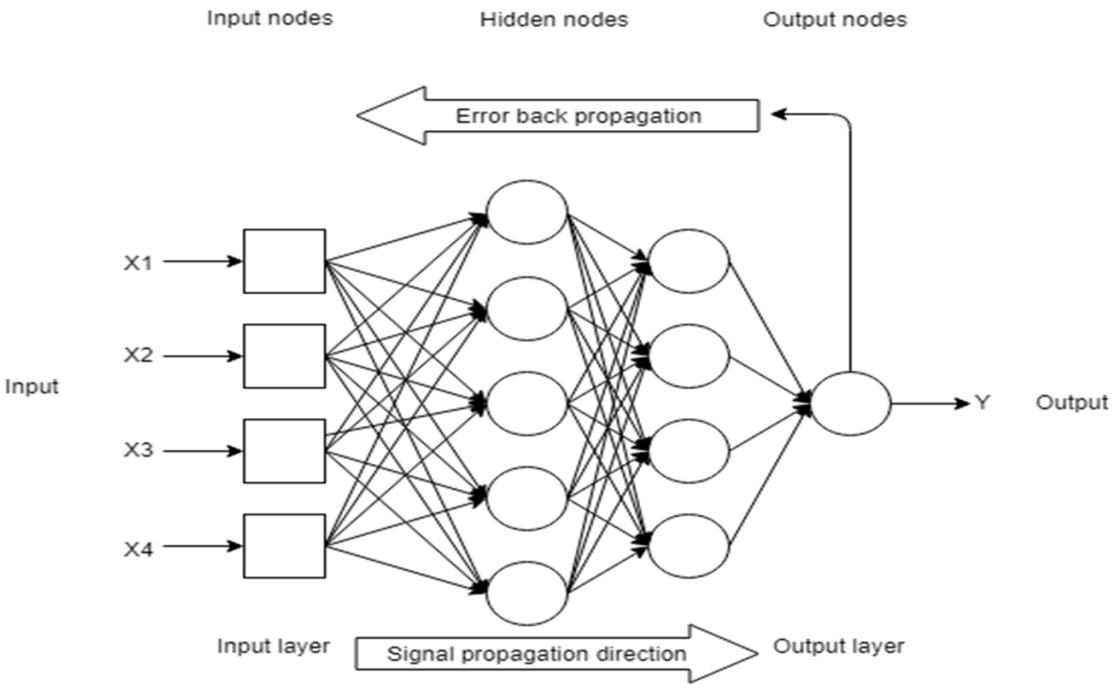


Figura 4 – Mecanismo de *backpropagation* em RNA. Fonte: (AZLAH et al., 2019).

Na segunda etapa, que pode ser chamada de *backward pass*, o algoritmo mede o erro de saída da rede, como por exemplo *Mean Squared Error* (MSE) (GÉRON, 2017), que pode ser ilustrada como a diferença entre a saída desejada e a saída real. A partir desse erro, é calculado quanto cada neurônio contribuiu para o erro dos neurônios na última camada oculta. Esse processo é repetido retroativamente, medindo a contribuição de erro de cada neurônio na camada oculta anterior, e assim por diante, até que o algoritmo alcance a camada de entrada.

Essa passagem reversa eficiente calcula o gradiente de erro em relação a todos os pesos de conexão na rede, propagando o gradiente de erro na rede para trás, o que dá origem ao nome do algoritmo.

A seguir, será exemplificado os processos descritos anteriormente, desde o *feed-forward* até o *backpropagation* na rede neural da Figura 5. Suponha que os neurônios utilizem a função de ativação sigmoide (2.4) para realizar o *forward pass* e o *backward pass*. Suponha também que a saída real Y seja 0.5 e a taxa de aprendizado representada por (η) seja 1. Na primeira etapa do *forward pass*, calculamos os valores das camadas H_1, H_2 e O_1.

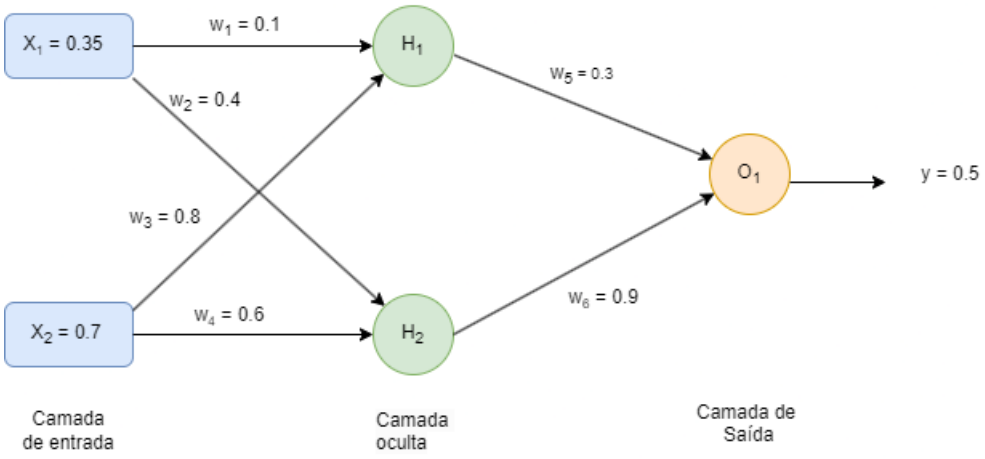


Figura 5 – Valores inicias da RNA. Fonte: Autor.

Para encontrar H_1, consideramos suas arestas de entrada e seus respectivos pesos: No nó h_1,

$$H_1 = (w_1 x_1) + (w_3 x_2) = (0.1 \cdot 0.35) + (0.8 \cdot 0.7) = 0.59$$

Aplicando a função de ativação (2.4) em h_1:

$$output_{h1} = F(0.59) = \frac{1}{1 + e^{-0.59}} = 0.64$$

Da mesma forma, são calculadas os valores de H_2 e O_1:

$$H_2 = (w_2 \cdot x_1) + (w_4 \cdot x_2) = (0.4 \cdot 0.35) + (0.6 \cdot 0.7) = 0.56$$

$$output_{h2} = F(0.56) = \frac{1}{1 + e^{-0.56}} = 0.63$$

A camada de saída final será:

$$O_1 = (w_5 \cdot y_1) + (w_6 \cdot y_2) = (0.3 \cdot 0.64) + (0.9 \cdot 0.63) = 0.75$$

$$output_{o1} = F(0.75) = \frac{1}{1 + e^{-0.75}} = 0.67$$

Note que a saída real é 0.5, mas obtivemos 0.67. Assumindo que seriam utilizados dois exemplos de treinamentos, o erro é calculado usando a fórmula do *Mean Squared Error* (MSE):

$$E_{\text{total}} = \sum \frac{1}{2}(target - output)^2$$

Desta forma, o cálculo do erro para o output O_1 será:

$$E_{\text{total}} = \frac{1}{2}(0.5 - 0.67)^2 = 0.0289$$

Após o *forward pass*, calculamos o *backpropagation* para atualizar os pesos para cada exemplo de treinamento. Começamos pela camada de saída, atualizando w_5 e w_6, calculamos a contribuição de w_5 para o E_{total} da seguinte forma:

$$\frac{\partial E_{\text{total}}}{\partial w_5} = \frac{\partial E_{\text{total}}}{\partial \text{output}_{o_1}} \times \frac{\partial \text{output}_{o_1}}{\partial O_1} \times \frac{\partial O_1}{\partial w_5} \tag{2.5}$$

De maneira simplificada, temos:

$$\frac{\partial E_{\text{total}}}{\partial \text{output}_{o_1}} = \text{output}_{o_1} - \text{target}_1 \tag{2.6}$$

$$\frac{\partial \text{output}_{o_1}}{\partial O_1} = \text{output}_{o_1}(1 - \text{output}_{o_1}) \tag{2.7}$$

$$\frac{\partial O_1}{\partial w_5} = \text{output}_{h1} \tag{2.8}$$

Portanto, a derivada final é dada por:

$$\frac{\partial E_{\text{total}}}{\partial w_5} = [\text{output}_{o1} - \text{target}_1] * [\text{output}_{o_1}(1 - \text{output}_{o_1})] * [\text{output}_{h1}]$$

E assim, podemos calcular o w_5 que será:

$$\frac{\partial E_{\text{total}}}{\partial w_5} = 0.17 * 0.22 * 0.64 = 0.02$$

E teremos o novo valor para w_5, ou seja new_w_5, que é dado por:

$$new\_w_5 = w_5 - \eta * \frac{\partial E_{\text{total}}}{\partial w_5}$$
$$new\_w_5 = 0.3 - 1 \times 0.02 = 0.28$$

De maneira similar ao w_5, calculamos o w_6, a única mudança seria no último componente da equação 2.8, onde neste caso para o w_6 o valor será $output_{h2}$.

E assim, calculamos o novo peso para w_6:

$$\frac{\partial E_{\text{total}}}{\partial w_6} = 0.17 * 0.22 * 0.63 = 0.02$$

$$new\_w_6 = 0.9 - 1 \times 0.02 = 0.88$$

Para calcular os pesos da camada oculta, propagamos os erros da camada de saída de volta para a camada oculta e ajustamos os pesos. A contribuição de w_1 para o erro total E_{total} é:

$$\frac{\partial E_{\text{total}}}{\partial w_1} = \frac{\partial E_{\text{total}}}{\partial \text{output}_{o_1}} \times \frac{\partial \text{output}_{o_1}}{\partial O_1} \times \frac{\partial O_1}{\partial \text{output}_{h_1}} \times \frac{\partial \text{output}_{h_1}}{\partial H_1} \times \frac{\partial H_1}{\partial w_1}$$

Substituindo as derivadas parciais:

$$\frac{\partial E_{\text{total}}}{\partial w_1} = [\text{output}_{o1} - \text{target}_1] * [\text{output}_{o_1}(1 - \text{output}_{o_1})] * w_5 * [\text{output}_{h1}(1 - \text{output}_{h1})] * x_1$$

Agora, podemos calcular o w_1 e atualizar o peso w_1:

$$\frac{\partial E_{\text{total}}}{\partial w_1} = 0.17 \times 0.22 \times 0.3 \times 0.23 \times 0.35 = 0.00090321$$

$$\text{new}\_w_1 = w_1 - \eta \times \frac{\partial E_{\text{total}}}{\partial w_1}$$

$$\text{new}\_w_1 = 0.1 - 1 \times 0.00090321 = 0.0990$$

Repetimos o processo para w_2, w_3 e w_4, ajustando os valores das variáveis em cada etapa, resultando nos novos valores:

$$\text{new}\_w_2 = 0.3972, \quad \text{new}\_w_3 = 0.7982, \quad \text{new}\_w_4 = 0.5946$$

Desta forma, aplicando o *Backpropagation*, os novos pesos são calculados e a nossa figura anterior (Figura 4) é atualizada com esses novos valores, conforme ilustrado na Figura 6.

2.2 Fundamentos sobre a Representações de Palavras com Redes Neurais Artificiais

Word Embeddings é uma técnica que representa palavras em vetores de valores reais em um espaço vetorial. Essa representação permite que palavras com significados semelhantes compartilhem vetores similares. Essa abordagem é considerada um avanço

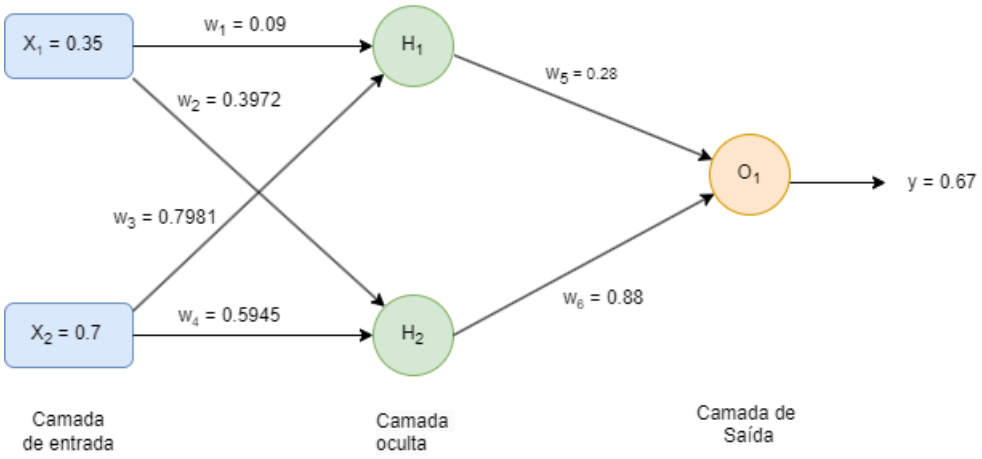


Figura 6 – Pesos atualizados após aplicar o *Backpropagation* Fonte: Autor.

importante na área de redes neurais em processamento de linguagem natural (PLN) (AUBAID; MISHRA, 2018).

A Figura 7 ilustra essa ideia, mostrando como as palavras estão relacionadas semanticamente, como "cachorro", "gato" e "cavalo", estão próximas umas das outras no espaço vetorial, enquanto palavras semântica e sintaticamente diferentes, como "sofá" e "mesa", estão distantes umas das outras, porém, próximas em relação a outras palavras relacionadas a móveis.

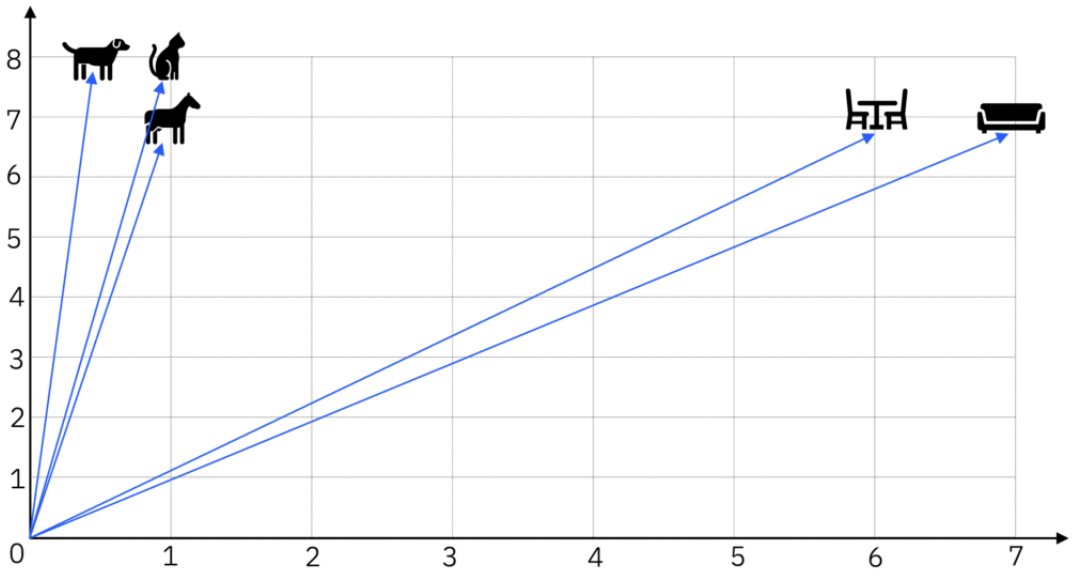


Figura 7 – Diversas palavras como *Embeddings* ocupando o espaço vetorial. Fonte: [1]

Os *Word Embeddings* capturam relações semânticas entre palavras em um espaço vetorial, como demonstrado na Figura 7. Esses vetores são representações obtidas a partir dos pesos em redes neurais treinadas prever palavras em sentenças. Existem duas abordagens principais para a geração de *embeddings*: estáticos e contextuais (dinâmicos).

[1] <https://brains.dev/2024/token-e-embedding-conceitos-da-ia-e-llms/>

Os *embeddings* estáticos são gerados com base em grandes conjuntos de dados textuais e capturam informações semânticas e sintáticas das palavras. Eles são fixos e não dependem do contexto em que as palavras aparecem. Algoritmos populares para geração de *embeddings* estáticos incluem:

- **Word2Vec**: Desenvolvido por Mikolov et al. (MIKOLOV et al., 2013), o *Word2Vec* é um dos métodos mais populares para geração de *embeddings* estáticos. Ele mapeia palavras para vetores em um espaço vetorial de alta dimensão, de forma que palavras com significados semelhantes fiquem próximas no espaço vetorial.
- **GloVe** (Global Vectors for Word Representation): Proposto por Pennington et al. (PENNINGTON; SOCHER; MANNING, 2014), o *GloVe* é outro método popular para geração de *embeddings* estáticos. Ele utiliza a co-ocorrência de palavras em grandes conjuntos de dados textuais para capturar as relações semânticas entre palavras.
- **FastText**: Introduzido por Joulin et al. (JOULIN et al., 2016), o *FastText* é uma extensão do *Word2Vec* que leva em consideração a estrutura morfológica das palavras. Ele é capaz de gerar *embeddings* para palavras que não estão presentes no vocabulário original, descompondo-as em subpalavras.

Por exemplo, considerando os *embeddings* gerados pelo *Word2Vec*, palavras como *"king"* e *"queen"* estarão próximas uma da outra no espaço vetorial, assim como *"man"* e *"woman"*, *"country"* e *"capital"*, entre outras relações semânticas, conforme apresentado na Figura 8.

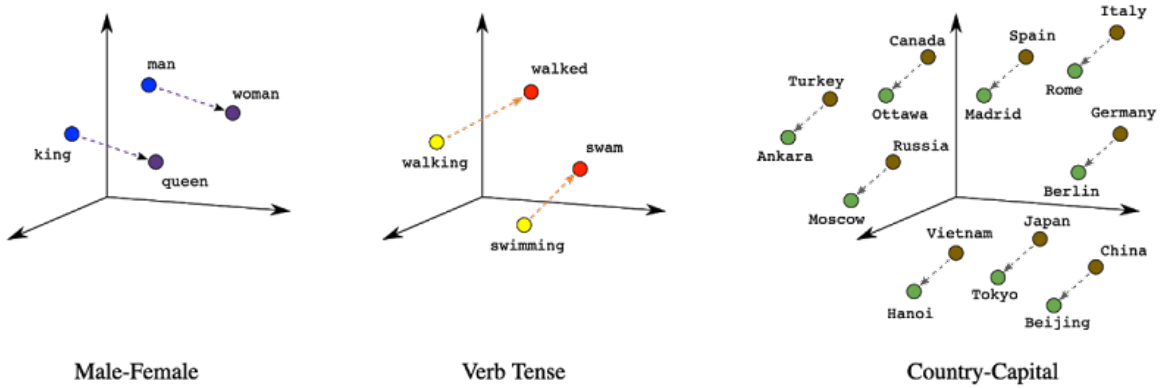


Figura 8 – Representação do *Word2Vec* em um espaço tridimensional. Fonte: Google Cloud. [2]

[2] <https://cloud.google.com/blog/topics/developers-practitioners/meet-ais-multitool-vector-embeddings/>

Por outro lado, os *embeddings* contextuais consideram o contexto em que as palavras aparecem em um texto. Diferentemente dos *embeddings* estáticos, os *embeddings* contextuais são calculados dinamicamente, levando em conta o contexto da frase em que a palavra aparece. Isso permite que o significado da palavra varie dependendo do contexto. Modelos populares de *embeddings* contextuais incluem:

- **ELMo (Embeddings from Language Models)**: Desenvolvido por Peters et al. (PETERS et al., 2018), o ELMo utiliza redes neurais bidirecionais para capturar o significado de uma palavra em relação ao seu contexto em uma frase.
- **GPT (Generative Pre-trained Transformer)**: Proposto por Radford et al. (RADFORD et al., 2018), o GPT é um modelo de linguagem baseado em *transformers* que captura o contexto de uma palavra em uma frase através de um mecanismo de atenção.
- **BERT (Bidirectional Encoder Representations from Transformers)**: Introduzido por Devlin et al. (DEVLIN et al., 2019), o BERT é um modelo de linguagem baseado em *transformers* que utiliza uma abordagem bidirecional para capturar o contexto de uma palavra em uma frase.

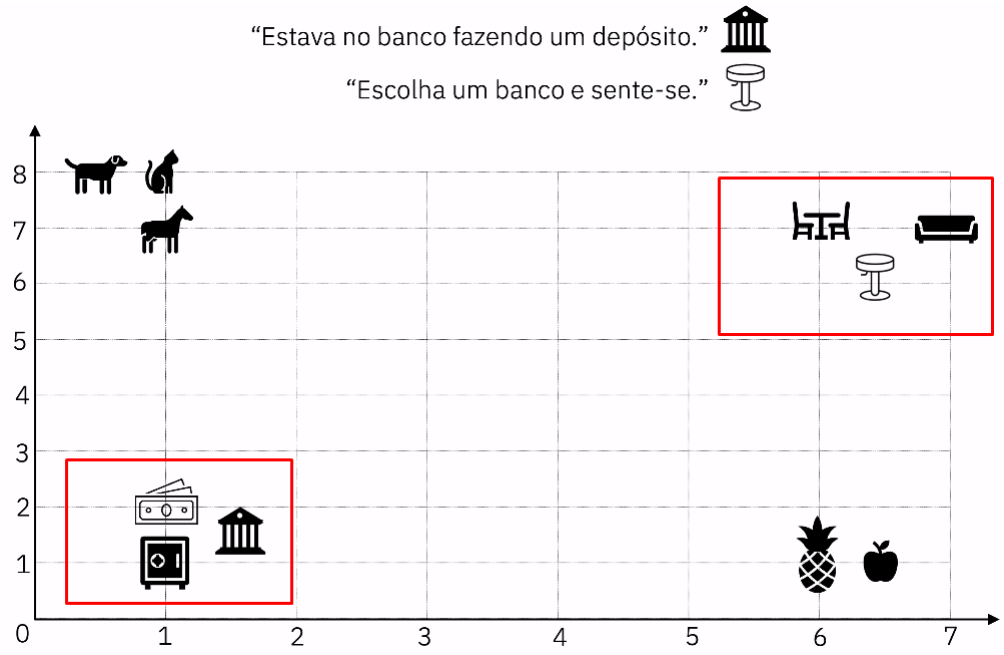


Figura 9 – Exemplo de *embeddings* contextuais para as palavras “banco”. Fonte: [3]

Para exemplificar esta ideia dos *embeddings* contextuais, considere a palavra “banco”, conforme mostrada na Figura 9. Dependendo do contexto em que essa palavra aparece em uma frase, seu significado pode variar. Se a palavra “banco” aparece em uma frase sobre

[3] <https://brains.dev/2024/token-e-embedding-conceitos-da-ia-e-llms/>

finanças, como “Estava no banco fazendo um depósito”, seu significado está relacionado a uma instituição financeira. No entanto, se a palavra “banco” aparece em uma frase como “Escolha um banco e sente-se”, seu significado está relacionado a um assento.

A Figura 9 ilustra como as palavras “banco”, têm significados diferentes dependendo do contexto em que são usadas em uma frase. No espaço vetorial dos *embeddings* contextuais, as representações dessas palavras variam com base no contexto em que aparecem.

Assim, esses modelos de *embeddings* são amplamente utilizados em várias tarefas de PLN, como classificação de texto, análise de sentimentos, tradução automática, entre outras, pois ajudam a capturar o significado e a semântica das palavras em um texto de maneira mais eficiente e precisa (HEIMERL; GLEICHER, 2018). Em nosso trabalho, o foco será centrado nos *Embeddings* contextuais. Na próxima seção, detalharemos como esses *embeddings* são gerados pela arquitetura *Transformers*.

2.3 Transformers

Os *transformers* são uma arquitetura de rede neural que revolucionou o campo do processamento de linguagem natural (PLN) desde sua introdução em 2017 proposta por (VASWANI et al., 2017). O *transformer* é uma rede neural com arquitetura *encoder-decoder* baseada em um mecanismo de atenção que aprende as relações contextuais entre palavras em um texto. A rede recebe uma sequência de palavras como entrada, codifica-as em representações nas camadas de atenção e as decodifica em palavras novamente. Nas próximas subseções, será detalhado cada bloco que compõe esta arquitetura.

2.3.1 Encoder e Decoder

A arquitetura do *Transformer* é dividida em duas partes: o *Encoder* e o *Decoder* (VASWANI et al., 2017). O *Encoder* mostrado na Figura 10 (metade à esquerda), consiste em uma pilha de camadas idênticas. Cada camada contém duas subcamadas: um mecanismo de *Multi-head Attention* e uma simples *Feed Forward Neural Network* (também conhecida como MLP). Conexões residuais e normalização de camada são aplicadas em cada subcamada. Todas as subcamadas, assim como as camadas de *embeddings*, produzem saídas de dimensão $d_{\text{model}} = 512$.

Similarmente, o *Decoder*, conforme ilustrado na Figura 10 (metade à direita), é constituído por uma pilha de camadas idênticas. Além das duas subcamadas presentes em cada camada do codificador, o decodificador integra uma terceira subcamada. Esta terceira subcamada realiza *Multi-head Attention* sobre a saída do codificador, possibilitando que o decodificador focalize diferentes partes da entrada durante o processo de decodificação. Conexões residuais e normalização de camada são aplicadas após cada subcamada. Ademais,

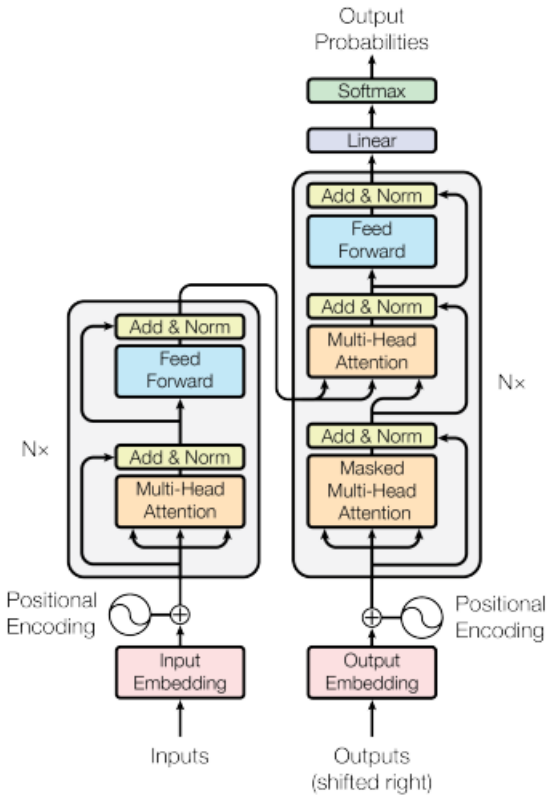


Figura 10 – Arquitetura do modelo *transformer*. Fonte: (VASWANI et al., 2017)

o decodificador incorpora uma camada de *Masked Multi-Head Attention* (VASWANI et al., 2017).

Podemos exemplificar o processo de *encoder* e *decoder* em uma aplicação de tradução de idioma, conforme ilustrado na Figura 11.

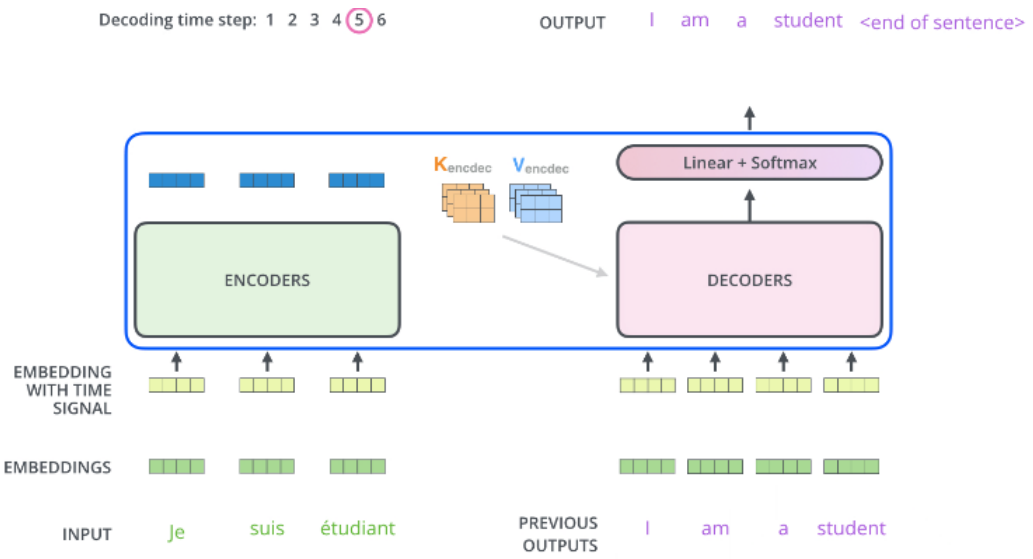


Figura 11 – *Encoder* e *Decoder* em uma aplicação de tradução. Fonte: [4]

[4] <https://jalammar.github.io/illustrated-transformer/>

Nesse exemplo da Figura 11, o *encoder* é responsável por receber uma frase em um idioma, processando os *embeddings* de entrada, enquanto o *decoder* utiliza esses *embeddings* para retornar a tradução correspondente, gerando os *embeddings* de saída. No caso apresentado, o modelo traduz uma frase do francês *("Je suis étudiant")* para o inglês *("I am a student")*.

De maneira geral, cada palavra da sentença de entrada gera um *embedding*, tanto para o encoder quanto para o decoder. A sentença completa entra no encoder, onde cada palavra é convertida em um embedding individual, além de ser gerado um *embedding* para a sentença inteira. Esses *embeddings* são processados pelo *encoder*, resultando em uma representação vetorial que captura o contexto e a relação entre as palavras.

Essa representação vetorial é então passada para o *decoder*, fornecendo a informação contextual necessária para a geração das palavras de saída. O *decoder* começa gerando a primeira palavra, utilizando essa representação. Cada palavra gerada é retroalimentada no *decoder* para a próxima etapa, combinando a representação vetorial da sentença com as palavras já geradas para prever a próxima palavra. O processo se repete até que um símbolo especial sinalize o fim da saída. No *decoder*, cada palavra gerada recebe um *embedding*, combinado com os *positional encodings*, cuja discussão será apresentada posteriormente, para preservar a sequência correta das palavras.

2.3.2 Embeddings

Outro componente da arquitetura do transformer são os *Embeddings*, onde estão representadas na Figura 10 como *input embeddings*. O embedding permite que um modelo aprenda sobre as relações entre palavras, tokens ou outras entradas, transformando dados de um espaço de alta dimensão para um espaço de baixa dimensão. Como destacado por Goldberg e Hirst (2017), a adoção de vetores densos e de baixa dimensionalidade simplifica a computação, uma vez que muitas plataformas de redes neurais podem não ser compatíveis com vetores dispersos e de alta dimensão.

Em PLN, a tokenização divide o texto de entrada em subunidades chamadas tokens, e são usados em etapas subsequentes, como análise morfológica e rotulagem de classe de palavras. Como essas etapas operam em sentenças individuais, a tokenização também identifica limites de sentenças e tokens (GREFENSTETTE, 1999). Os tokens são derivados de um corpus[5] de dados que pode conter capítulos, parágrafos ou sentenças, com a tokenização mais comum sendo por palavra. Todas as palavras únicas formam o vocabulário e, geralmente, cada palavra é associada a um número inteiro para facilitar o processamento computacional.

A figura 12 demonstra esse processo de decomposição de um corpus maior em seus

[5] Um corpus é um banco de textos usado para treinar, avaliar ou estudar modelos de linguagem.

componentes e a atribuição de números inteiros a cada um.

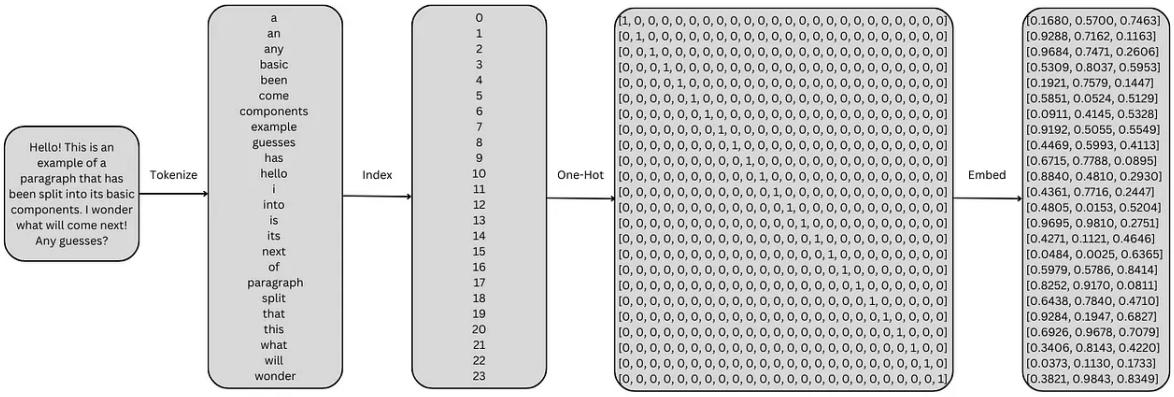


Figura 12 – Exemplo básico de Embedding. Fonte: [6]

No início de PLN, números inteiros eram utilizados para criar vetores *one-hot encoding* para cada palavra, onde apenas um elemento era ativado, definido como 1, enquanto todos os outros eram desativados, definidos como 0, resultando em vetores ortogonais entre si. No entanto, essa abordagem, adotada nos primórdios do PLN, tornou-se impraticável para identificar a distância semântica entre as palavras (WANG et al., 2020). Para grandes coleções de textos, é preferível utilizar uma camada de embedding em vez de vetores *one-hot encoding*, embora estes ainda sejam úteis para representar tokens ou classes em modelos com um número limitado de opções.

Uma camada de embedding mapeia vetores one-hot para dimensões menores, permitindo que vetores densos e de baixa dimensão representem palavras em vez de vetores dispersos e de alta dimensão. Cada palavra é então representada por um vetor de dimensão d_{model}, que pode variar dependendo da tarefa, comumente utilizando valores como 256, 512 ou 1024. Este método facilita a aprendizagem de relações semânticas entre palavras e tokens em modelos de processamento de linguagem natural.

A matriz de embeddings tem tamanho $(vocab\_size, d\_model)$, permitindo que uma matriz one-hot de tamanho $(seq\_length, vocab\_size)$ seja multiplicada por ela para obter uma nova representação *embedded*. Aqui, seq_length é o número de tokens em uma sequência. Na prática, seria utilizado um subconjunto do vocabulário, como "a basic paragraph", que seria tokenizado, indexado e convertido em uma matriz *one-hot*. Esses vetores *one-hot* seriam então multiplicados pela matriz de *embeddings*. Essas etapas são demonstradas na Figura 13.

[6] <https://medium.com/@hunter-j-phillips/the-embedding-layer-27d9c980d124/>
[7] <https://medium.com/@hunter-j-phillips/the-embedding-layer-27d9c980d124/>

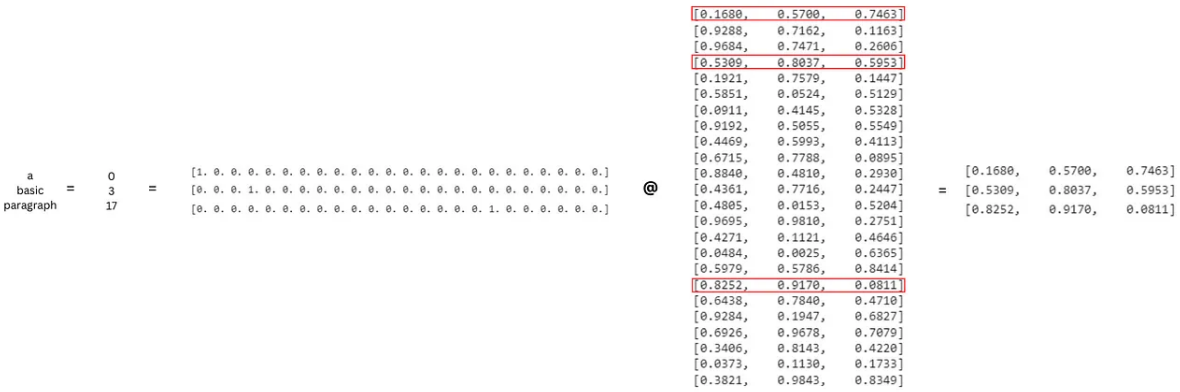


Figura 13 – Exemplo básico de Embedding. Fonte: [7]

Uma sequência *embedded* resultaria em um tamanho (seq_length, d_model), representando cada palavra por um vetor de dimensão d_model em vez de um vetor *one-hot* de $vocab\_size$ elementos. Por exemplo, uma sequência indexada de forma $(3, 24)$ multiplicada por uma matriz de embeddings de forma $(24, 3)$ resulta em uma matriz $(3, 3)$, onde cada palavra é representada por seu vetor de embeddings de 3 elementos, conforme apresentado na Figura 13.

Nesta Figura 13, a multiplicação de uma matriz one-hot por uma matriz de *embedding* ilustra a abordagem adotada na camada de *input embedding* representa na Figura 10 em representações gerais de *transformers*. Esse processo retorna os vetores de embedding correspondentes sem alterações. Especificamente, quando uma matriz one-hot (que representa palavras ou tokens como vetores binários) é multiplicada por uma matriz de embedding (que contém os vetores de embeddings dos *tokens*), o resultado é o vetor de *embedding* do *token* representado pela matriz one-hot.

Essa abordagem, usada nas camadas de *input embeddings* dos modelos de redes neurais, converte vetores *one-hot* esparsos e de alta dimensionalidade em representações densas de menor dimensionalidade que capturam informações semânticas e sintáticas dos *tokens*. Isso facilita o processamento subsequente nas camadas seguintes da rede neural, melhorando a eficiência e o desempenho do modelo.

Em *transformers*, a camada de *embedding* é usada no codificador e no decodificador (VASWANI et al., 2017). A única adição ao módulo *Embedding* é um produto escalar. Os pesos de *embedding* são multiplicados por $\sqrt{d_{\text{model}}}$. Isso ajuda a preservar o significado subjacente quando o *embedding* é adicionada ao *positional encoding* na próxima etapa. Isto essencialmente torna o *positional encoding* relativamente menor e diminui o seu impacto nas *embeddings*.

2.3.3 Positional Encoding

O *Positional Enconding* (PE) está presente na arquitera dos *Transformers*, conforme apresentado na Figura 10. O *Positional enconding* descreve a localização ou posição relativa ou absoluta de *tokens* em uma sequência, de modo que cada posição receba uma representação exclusiva (VASWANI et al., 2017). Os *transformers* usam um esquema de *Positional Enconding* inteligente, onde cada posição/índice é mapeado para um vetor. Essa é a fase inicial para a construção do *embedding* dinâmico final, visto que são gerados os *embeddings* posicionais para cada termo. Essa etapa é fundamental para a compreensão da ordem das palavras em uma sentença.

O *Positional Encoding* é essencial para atribuir a cada posição na sequência um vetor que codifica sua posição, permitindo ao modelo diferenciar entre diferentes posições e capturar a ordem dos elementos. Para garantir que essas representações sejam contínuas e variem suavemente ao longo da sequência, os valores atribuídos pelo *Positional Encoding* devem ser cíclicos (VASWANI et al., 2017).

Essa ciclicidade é alcançada utilizando funções seno e cosseno, que proporcionam uma codificação suave e repetitiva das informações de posição ao longo da sequência. Essa abordagem é particularmente útil em sequências longas, onde a distinção entre posições é crucial para a compreensão da estrutura da sequência.

Além disso, conforme mencionado por Vaswani et al. (2017), a versão com funções seno e cosseno é escolhida porque pode permitir que o modelo extrapole para comprimentos de sequência maiores do que aqueles encontrados durante o treinamento. Se um modelo puder identificar as posições relativas das palavras por meio dessas funções periódicas, ele deverá ser capaz de detectar quaisquer posições relativas, independentemente do comprimento da sequência.

Conforme abordado pelo autor Vaswani et al. (2017), é adicionado *positional enconding* aos *inputs embeddings* na parte inferior das pilhas do codificador e do decodificador, conforme apresentado na Figura 14.

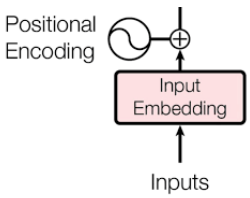


Figura 14 – *Positional Encoding*. Fonte: (VASWANI et al., 2017)

Os *positional encodings* possuem a mesma dimensão que os *embeddings* para permitir que ambos possam ser somados (VASWANI et al., 2017). Um exemplo dessa soma de vetores é apresentado na Figura 15. O *Transformer* utiliza *word embeddings* de 512

dimensões (elementos), e o *positional encoding* também utiliza 512 dimensões, garantindo que os dois vetores possam ser somados diretamente.

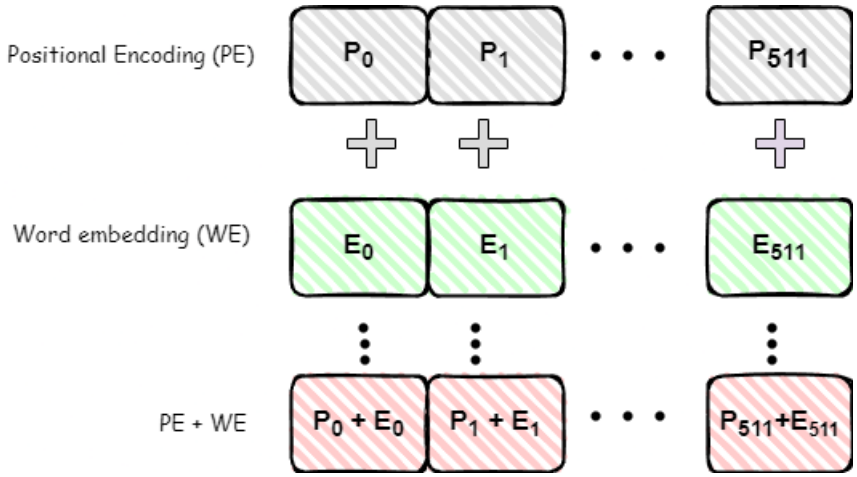


Figura 15 – *Positional Encoding* (PE), *Word Embedding* (WE) e a soma (PE + WE). . Fonte: Autor.

Portanto, cada elemento da sequência é enriquecido com sua posição relativa na sequência, permitindo que o modelo capture de forma mais precisa o contexto e a relação entre os elementos. Um exemplo dessa codificação é ilustrado na Figura 16.

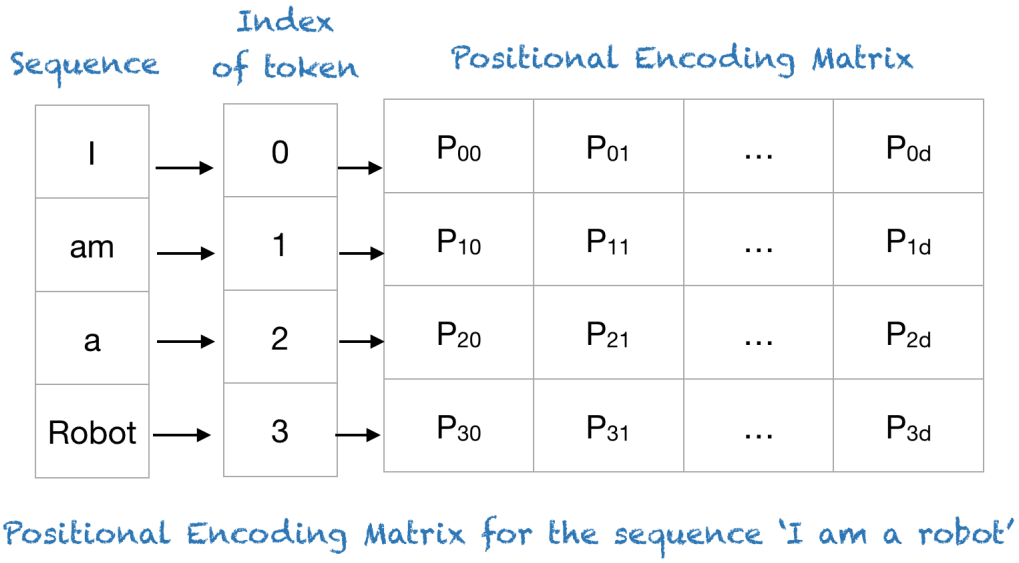


Figura 16 – *Positional Encoding*. Fonte: Jason Brownlee [8]

O *Positional Encoding* é realizada por meio de funções seno e cosseno de diferentes frequências, conforme apresentado a seguir:

$$\mathrm{P}_{(k,2i)} = \sin\left(\frac{k}{n^{2i/d}}\right)$$

[8] <https://machinelearningmastery.com/a-gentle-introduction-to-positional-encoding-in-transformer-models-part-1/>

$$P_{(k,2i+1)} = \cos\left(\frac{k}{n^{2i/d}}\right)$$

Onde:

- K: Posição de um objeto na sequência de entrada. $0 \leq K < \frac{L}{2}$;
- d: Dimensão do espaço de incorporação de saída;
- $P(k, j)$: Função de posição para mapear uma posição k na sequência de entrada (k, j) para indexar da matriz posicional;
- n: escalar definido pelo usuário, definido como 10.000 pelos autores (VASWANI et al., 2017);
- i: Usado para mapeamento de índices da coluna $0 \leq i < \frac{d}{2}$, com um único valor de i mapeia para funções seno e cosseno.

2.3.4 Attention

A *Attention* em *Transformers* é uma técnica usada para identificar quais termos de uma sequência são mais importantes dentro de uma sentença e como esses termos se relacionam com outros termos. Especificamente, a *Attention* calcula a importância relativa de cada termo em relação aos demais termos da sentença, permitindo que o modelo foque nas partes mais relevantes do contexto ao processar cada palavra.

De acordo com o autor Vaswani et al. (VASWANI et al., 2017), uma função de atenção pode ser descrita como o mapeamento de uma consulta e um conjunto de pares chave-valor para uma saída, onde a consulta, as chaves, os valores e a saída são todos vetores. A saída é calculada como a soma ponderada dos valores, em que o peso atribuído a cada valor é determinado por uma função de compatibilidade entre a consulta e a chave correspondente.

Um bloco de *attention*, divide o texto e associa cada *token* ou palavra a um vetor. Além de refinar o significado das palavras individualmente, o bloco de atenção possibilita que o modelo mova informações codificadas em uma *embedding* para outra, permitindo a consideração de informações mais amplas e complexas do que apenas o significado de uma única palavra.

Descreveremos em detalhes as representações vetoriais de *Query*, *Key* e *Value*. Antes de prosseguir com a detalhes destas representações, é importante definir as notações utilizadas que serão abordadas nas seções seguintes. A seguir, são definidos as notações de representações vetoriais de *Embeddings* e matrizes de *Embeddings* usadas nesta seção:

- $\underset{E_i}{\rightarrow}$: Esta notação representa o vetor de *embedding* associado ao token i (índice da posição do termo na sentença);
- $\underset{Q_i}{\rightarrow}$: Refere-se ao vetor de consulta (*Query*) para o token i;
- $\underset{K_i}{\rightarrow}$: O vetor de chave (*Key*) associado ao token i;
- $\underset{V_i}{\rightarrow}$: Representa o vetor de valor (*Value*) para o token i;
- $\underset{W_q}{\rightarrow}$: Esta notação representa a matriz de consulta (*Query matrix*), sendo a matriz de pesos aprendíveis para as *Queries*. Quando um *Embedding* E_i correspondente a um termo na sentença é multiplicado por esta matriz, o resultado da multiplicação corresponde a um novo *Embedding* que se refere a um vetor de consulta Q_i;
- $\underset{W_k}{\rightarrow}$: Representa a matriz de chave (*Key matrix*), sendo a matriz de pesos aprendíveis para as *Keys*. Quando um *Embedding* E_i correspondente a um termo da sentença, é multiplicado por esta matriz, o resultado da multiplicação corresponde a um novo *Embedding* que se refere a um vetor de chave K_i;
- $\underset{W_v}{\rightarrow}$: Refere-se a matriz de valor (*Value matrix*), sendo a matriz de pesos aprendíveis para os *values*. Quando um *Embedding* E_i correspondente a um termo E_i é multiplicado por esta matriz, o resultado da multiplicação corresponde a um novo *Embedding* que se refere a um vetor de valores V_i.

A *Query* em um bloco de *Attention* é um vetor que representa a solicitação de informações específicas sobre uma palavra ou token em uma sequência. Sua função é essencial para determinar a importância relativa de outras palavras ou tokens em relação à palavra ou token em questão.

Por exemplo, conforme demonstrado na Figura 17, suponha que um bloco de *attention* seja capaz de analisar a informação de adjetivos na sentença. A *Query* pode ser concebida com o intuito de extrair informações relacionadas à natureza descritiva do adjetivo, sua interação com outros termos na sentença e sua influência no significado global da sentença.

É relevante salientar que a referência ao termo "adjetivo" tem caráter meramente exemplificativo, evidenciando que a *Query* pode ser direcionada a qualquer atributo ou característica semântica pertinente à tarefa em questão. Além disso, é importante ressaltar que essa representação não está restrita a uma classe gramatical específica, podendo abranger uma ampla gama de atributos, tais como gênero, polaridade ou quaisquer outros aspectos relevantes ao contexto em análise.

[9] Disponível em: <https://www.3blue1brown.com/lessons/attention>

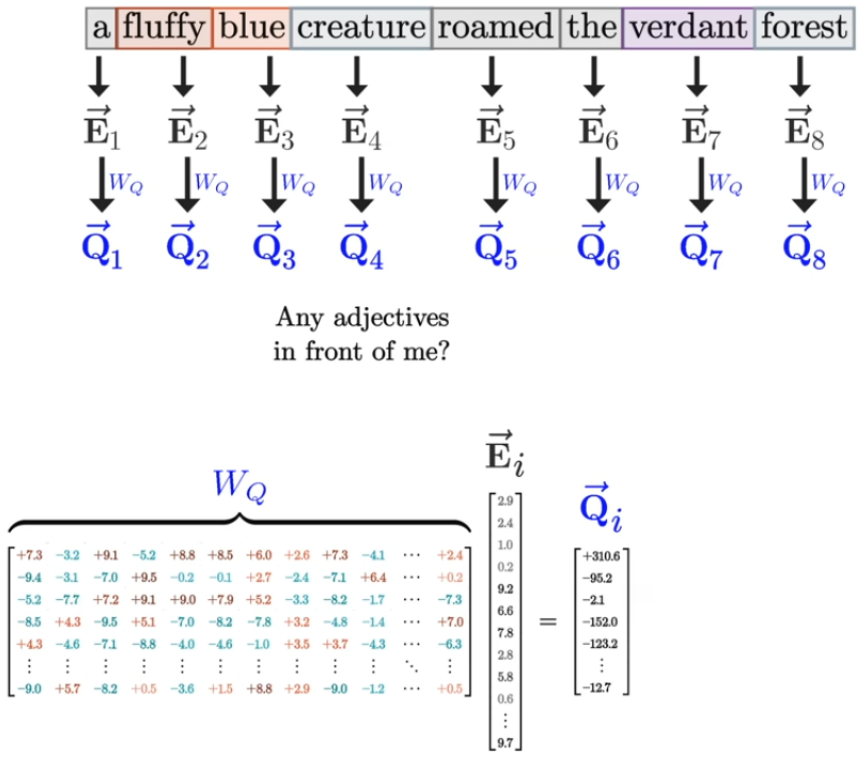


Figura 17 – Exemplo de *query*. Fonte: 3Blue1Brow. [9]

O cálculo da *Query* (Q_i) para cada palavra é realizado da seguinte maneira: O vetor do *Embedding* (E_i) é multiplicado por uma matriz de consulta (W_q) para obter o vetor de consulta final (Q_i), conforme representado pela seguinte fórmula: $W_q \times E_i = Q_i$. Isso nos dá um vetor de consulta para cada palavra no contexto.

Em seguida, o vetor de *Keys*, representada por K, é multiplicada por todas as *embeddings* E, a fórmula para calcular é descrita como: $W_k \times E = K$, gerando uma sequência de vetores chamada *keys*. Conceitualmente, podemos pensar nas *keys* como vetores que contêm a informação necessária para responder a uma *query* em um determinado contexto. As *keys* são, essencialmente utilizadas para ajudar o modelo a identificar a relevância da informação quando uma *query* é feita.

Conforme ilustrado na Figura 18, essa correspondência é medida calculando o produto escalar entre cada par de *key* e *query*. Essa operação resulta em pontuações que indicam o quanto cada *key* é relevante para uma *query* em específico.

Por fim, temos o *Value*, que é essencial para extrair a informação original dos dados. Os *values* são representações vetoriais que contêm a informação efetiva que será recuperada e combinada para formar a saída final do mecanismo de *Attention*. Eles encapsulam os dados cruciais necessários para a tarefa de atenção.

Para realizar essa transformação, usamos uma terceira matriz chamada *Value*

[10] <https://www.3blue1brown.com/lessons/attention>

Figura 18 – Exemplo de *key*. Fonte: 3Blue1Brow. [10]

Matrix, representada por W_v. Essa matriz de pesos é aplicada às embeddings de entrada E para gerar os *values* V. A fórmula para calcular os *values* é: $W_v \times E = V$, onde V representa a matriz resultante dos *values*.

Após ter o cálculo destas matrizes apresentadas anteriormente, é calculado os produtos escalares das *queries* (Q) com todas as *keys* (k), e cada um é dividido por $\sqrt{d_k}$, e em seguida, aplica-se uma função Softmax para obter os pesos dos valores (VASWANI et al., 2017). Na prática, as consultas são processadas simultaneamente, agrupadas em uma matriz Q, enquanto as chaves e os valores são agrupados nas matrizes K e V, respectivamente. É calculado a matriz de saídas da seguinte forma:

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

Os autores Vaswani et al. (2017) explicam a razão pela qual o produto escalar entre as *queries* e as *keys* é dividido por $\sqrt{d_k}$. Eles demonstram que introduzindo um fator de escala de $\sqrt{\frac{1}{d_k}}$, é possível contrabalançar o aumento da magnitude dos produtos ponto para valores maiores de d_k. Essa normalização desempenha um papel crucial na manutenção da estabilidade do processo de treinamento, garantindo que a função softmax não seja empurrada para regiões com gradientes extremamente pequenos.

2.3.5 Multi-Head Attention

Conforme abordado por Vaswani et al. (2017), em vez de utilizar uma única função de atenção, projetamos linearmente as consultas, chaves e valores em várias dimensões diferentes. Isso nos permite realizar a atenção de forma paralela, proporcionando maior capacidade de modelagem.

A operação de atenção multi-cabeça (*Multi-Head*) concatena os resultados de várias cabeças de atenção diferentes (*h heads*). Em cada cabeça de atenção, realizamos uma transformação linear das consultas, chaves e valores originais, seguida pela aplicação da função de atenção.

Essa abordagem permite que o modelo aprenda diferentes representações dos dados de entrada, capturando informações relevantes em múltiplos subespaços de representação. A intuição por trás do *Multi-Head Attention* pode ser vista na Figura 19, onde são separadas as matrizes de pesos Q, K, V para cada *head*, resultando em diferentes matrizes Q, K, V, respectivamente.

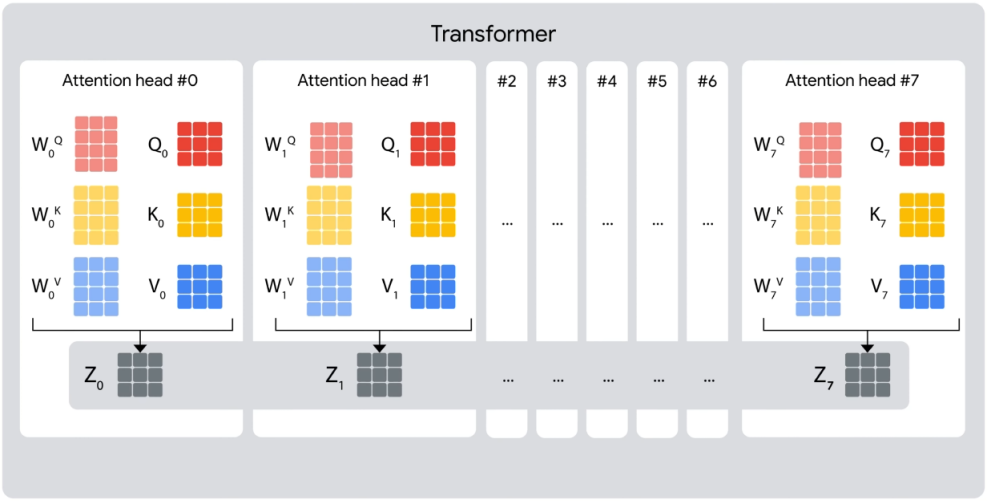


Figura 19 – Concatenação no *Multi-Head Attention*. Fonte: [11]

A formulação da *Multi-Head Attention* é representado a seguir como:

$$\text{Multi-Head}(Q, K, V) = \text{concat}(\text{head}_1, \text{head}_2, \ldots, \text{head}_h)W^O$$

onde,

$$\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$

[11] <https://goo.gle/3Wk3jnC>

Os valores resultantes de cada *head* de *attention* são concatenados e, em seguida, linearmente projetados novamente pela matriz de projeção W^O, para produzir os valores finais da operação de *Multi-head attention*.

2.3.6 Masked Multi-Head Attention

A subcamada de autoatenção no decodificador é adaptada para evitar que posições considerem posições futuras. Essa adaptação, combinada com o deslocamento das *Embeddings* de saída em uma posição, assegura que as previsões para uma determinada posição sejam baseadas exclusivamente nas saídas conhecidas em posições anteriores (VASWANI et al., 2017).

Essa etapa é denominada *Masked Multi-Head Attention*. Antes de aplicar a atenção, uma máscara é aplicada às chaves (*keys*) e valores (*values*) para ocultar informações futuras. A máscara pode ser implementada utilizando uma matriz contendo zeros e valores negativos infinitos, onde os valores negativos infinitos bloqueiam a atenção para as posições futuras. Essa abordagem garante que o modelo não utilize informações do futuro durante o treinamento, preservando assim a ordem temporal dos dados e garantindo coerência nas previsões de sequência.

2.3.7 Feed Forward Neural Network

Além das subcamadas de *Multi-Head Attention*, cada camada do *Encoder* e do *Decoder* no *Transformer* também inclui uma subcamada de *Feed Forward*, esta rede neural, também conhecida como MLP. No trabalho do *Transformer* (VASWANI et al., 2017), esta subcamada consiste em duas operações lineares separadas por uma função de ativação, geralmente uma função ReLU (*Rectified Linear Unit*). A função ReLU atribui zero a todos os valores de entrada negativos e mantém os valores positivos inalterados. A saída da subcamada de *Feed Forward* é calculada como:

$$\mathrm{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2 \tag{2.9}$$

Onde W_1, W_2, b_1 e b_2 são parâmetros da rede que são aprendidos durante o treinamento.

2.3.8 Residual conection

O codificador e o decodificador incorporam as conexões residuais (*residual connections*). Essas conexões consistem em somar a entrada de uma subcamada (*subLayer*) à sua saída, criando um caminho de atalho que facilita o fluxo de informação. Dessa forma, elas amenizam o problema do desaparecimento do gradiente em redes neurais profundas, além de tornar o treinamento mais estável e eficiente (HE et al., 2015).

Formalmente, se a entrada do bloco for denotada como x e a saída do bloco subsequente (como um bloco de autoatenção ou *feed-forward*) for um SubLayer(x), a operação de adição residual pode ser representada como:

$$y = x + \text{SubLayer}(x) \tag{2.10}$$

Nos modelos *Transformer*, as conexões residuais são empregadas em dois locais principais: após o Mecanismo de atenção e após a Camada *Feed-Forward*. No primeiro caso, a saída do mecanismo de autoatenção é somada à sua entrada original. No segundo caso, a saída da camada *feed-forward* é somada à sua entrada original antes de ser aplicada a normalização, que será explicada na próxima seção.

2.3.9 Add & Norm

Nos modelos *Transformer*, a etapa de Add & Norm é essencial tanto no codificador quanto no decodificador. Esse componente combina a adição residual e a normalização de camada para estabilizar e melhorar o processo de treinamento.

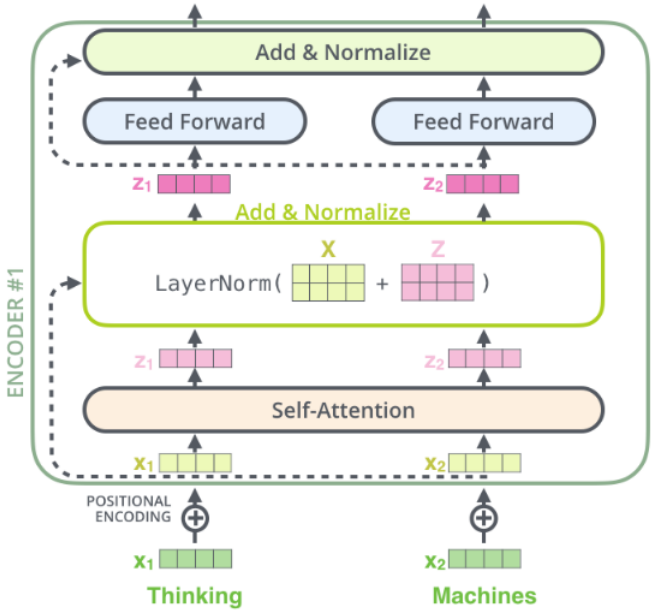


Figura 20 – Normalização de camada. Fonte: [12]

Primeiramente, é aplicada uma adição residual (*Add*), apresentada anteriormente na equação 2.10, após isso, aplica-se a normalização de camada (*Layer Normalization*) para estabilizar as ativações. A normalização de camada, aplicada após a conexão residual, ajusta a saída para ter média zero e variância unitária, contribuindo para a estabilidade e eficiência do treinamento (BA; KIROS; HINTON, 2016). Desta forma, a normalização é

[12] <https://jalammar.github.io/illustrated-transformer/>

definida da seguinte maneira:

$$\text{LayerNorm}(x + \text{Sublayer}(x))$$

A normalização de camada ajusta cada unidade de ativação para garantir que a saída tenha uma distribuição consistente, facilitando o treinamento de redes neurais profundas. Ao padronizar as ativações, ela mantém a qualidade das informações através das camadas, promovendo um aprendizado mais robusto e eficiente.

Para exemplificar e ilustrar essa ideia, considere a Figura 20. Neste bloco, na matriz X, cada linha desta matriz representa um *embedding* E_i, enquanto a matriz Z corresponde aos *embeddings* resultantes da camada de atenção. A operação envolve a soma das duas matrizes seguida pela aplicação da normalização de camada.

2.4 BERT

O BERT (*Bidirectional Encoder Representations from Transformers*) (DEVLIN et al., 2019) é um modelo de representação de linguagem que se baseia na arquitetura *transformer* (VASWANI et al., 2017). Ao contrário da arquitetura *transformer*, que emprega tanto codificador quanto decodificador, o BERT visa criar um modelo de linguagem, sendo assim, apenas o mecanismo codificador é utilizado.

Sua característica bidirecional deriva do fato de representar as palavras considerando tanto o contexto à esquerda quanto o contexto à direita, esta ideia é mostrado na Figura 21, onde a arquitetura do modelo BERT é composta por múltiplos codificadores *Transformers* bidirecionais, organizados em várias camadas.

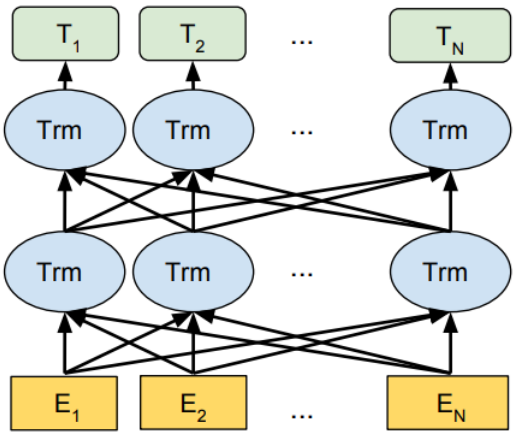


Figura 21 – Arquitetura do *BERT*. Fonte: (DEVLIN et al., 2019).

O BERT apresenta duas arquiteturas: O $BERT_{BASE}$ e $BERT_{LARGE}$, estas arquiteturas são apresentadas na Figura 22. O $BERT_{BASE}$ é formado por uma sequência de 12 codificadores, enquanto o $BERT_{LARGE}$ é composto por 24 codificadores empilhados.

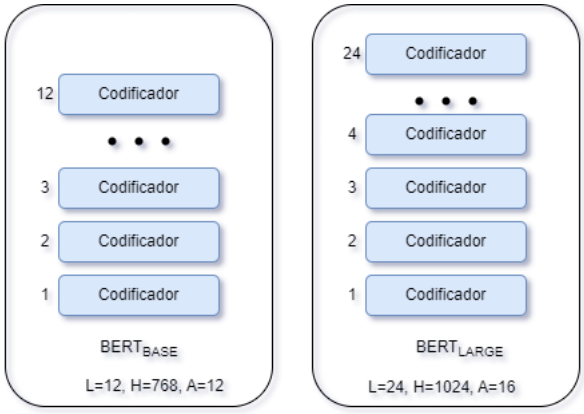


Figura 22 – Arquiteturas: $BERT_{BASE}$ e $BERT_{LARGE}$. Fonte: Autor.

O número de camadas, ou blocos *Transformer*, é denotado como L, o tamanho da camada oculta é denotado como H, e o número de *heads* de *attentions* é denotado como A. O modelo *BASE* possui 110 milhões de parâmetros, enquanto o modelo *LARGE* possui 340 milhões de parâmetros. (DEVLIN et al., 2019).

2.4.1 Pré-Treinamento do BERT

O modelo BERT original (DEVLIN et al., 2019) foi pré-treinado utilizando o BookCorpus (cerca de 800 milhões de palavras) e a Wikipedia em inglês (com aproximadamente 2,5 milhões de palavras). O BERT utiliza duas estratégia de treinamento: O *Masked Language Modeling* (MLM) e *Next Sentence Prediction* (NSP).

2.4.2 Masked Language Modeling (MLM)

Nesta etapa de *Masked Language Modeling* (MLM), Para treinar uma representação bidirecional, uma porcentagem dos tokens de entrada é aleatoriamente mascarada e, em seguida, esses tokens mascarados são previstos (DEVLIN et al., 2019). Os vetores finais ocultos desses tokens mascarados são então usados para prever palavras específicas, em vez de reconstruir toda a entrada.

Antes de alimentar sequências de palavras no BERT, 15% das palavras em cada sequência são substituídas pelo token [MASK]. Esta substituição é realizada da seguinte forma: em 80% dos casos, substitui-se pelo token [MASK], em 10% dos casos, substitui-se por um token aleatório, e nos 10% restantes, o token permanece inalterado. O modelo, então, tenta prever o valor original dessas palavras mascaradas com base no contexto fornecido pelas palavras não mascaradas na sequência. (DEVLIN et al., 2019).

Para ilustrar o conceito de MLM, a Figura 23 apresenta a estratégia de treinamento do MLM. Nesta figura, uma sentença é utilizada como entrada (Input Tokens), e um mascaramento aleatório é aplicado (Input Tokens Masked). No exemplo, a palavra na

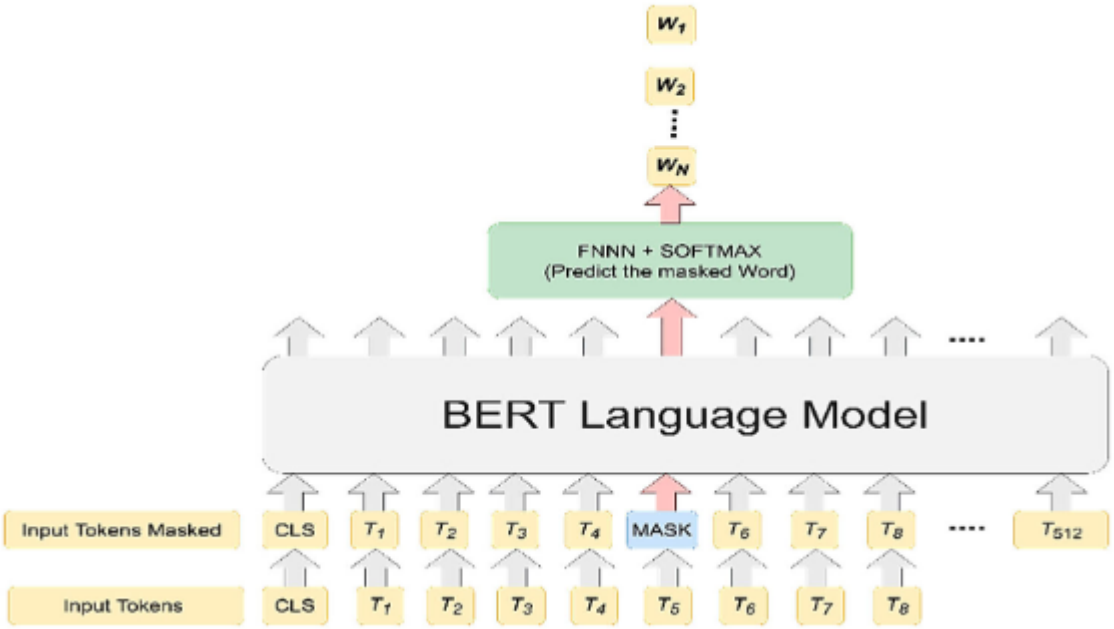


Figura 23 – Treinamento do *Masked Language Modeling* (MLM). Fonte: (OZCIFT et al., 2021).

posição T_5 foi substituída pelo token [MASK]. O objetivo do MLM é prever a palavra mascarada, T_5, com base no contexto da frase.

Para alcançar isso, uma camada totalmente conectada (feed forward neural network) é aplicada, seguida de uma função softmax, a qual gera a probabilidade de cada palavra no vocabulário. Assim, o MLM aborda um problema de classificação, retornando uma probabilidade para cada token do vocabulário, indicando a probabilidade de cada palavra ter sido a mascarada.

2.4.3 Next Sentence Prediction (NSP)

A segunda etapa do treinamento é o *Next Sentence Prediction* (NSP), que consiste em prever se uma determinada sentença é a próxima em uma sequência de texto ou não. Isso é feito em um formato binarizado, onde metade das vezes a sentença seguinte é a próxima no texto (rotulada como *IsNext*), e metade das vezes é uma sentença aleatória do corpus (rotulada como *NotNext*) (DEVLIN et al., 2019).

O NSP utiliza dois tokens especiais: [CLS] e [SEP]. O token [CLS] é um marcador de classificação binária inserido no início da primeira frase. Ele desempenha um papel crucial na previsão se a segunda frase é uma sequência lógica da primeira. Por outro lado, o token [SEP] é utilizado como um token de separação, indicando o fim de uma frase. Assim, a primeira frase é delimitada da segunda pelo token [SEP], permitindo ao modelo entender claramente onde uma frase termina e a próxima começa.

Conforme observado na Figura 24, os *embeddings* de entrada no modelo BERT

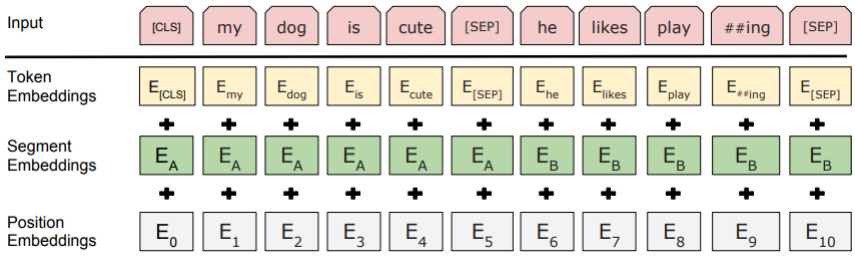


Figura 24 – Representação da entrada do BERT. Fonte: (DEVLIN et al., 2019).

são compostos pela soma dos *token embeddings* com os *segment embeddings* e os *position embeddings*. Essa composição é essencial para fornecer ao modelo informações sobre a estrutura e o contexto das entradas de texto.

Os *token embeddings* representam as palavras individuais em cada sequência de texto e capturam seu significado semântico. Além disso, o token especial [CLS] é inserido no início de cada sequência para fornecer ao modelo uma posição que contenha a representação do contexto da frase, em outras palavras, seu embedding é usado para representar o contexto geral da entrada. O token [CLS] é a informação utilizada para classificar se a próxima sentença é de fato a setença original ou uma sentença aleatória.

Os *segment embeddings* são adicionados para permitir que o modelo diferencie entre diferentes segmentos ou partes de texto em uma entrada composta, como duas frases em uma tarefa NSP. Por exemplo, o token [SEP] é inserido entre as duas sentenças em cada exemplo de entrada, e seus embeddings de segmento ajudam o modelo a entender a transição entre as sentenças.

Por fim, os *position embeddings* são incorporados para informar ao modelo a posição relativa de cada token na sequência de entrada, garantindo que ele entenda a ordem das palavras.

2.4.4 Fine-tuning

Na etapa de *fine-tuning*, utiliza-se uma rede pré-treinada, como o BERT, que foi treinado com *Masked Language Modeling* (MLM) e *Next Sentence Prediction* (NSP) (DEVLIN et al., 2019), e adiciona-se uma camada extra para a tarefa específica desejada. O *fine-tuning* pode ser definido da seguinte maneira: o modelo padrão do BERT é um classificador de sequência. Nesse contexto, o estado oculto final da primeira palavra ([CLS]) do BERT é introduzido em uma camada totalmente conectada para realizar a avaliação com softmax (MA, 2019).

O processo inicia-se com um modelo de *transformer* pré-treinado, cujos pesos já estão ajustados a partir de dados gerais. Para ilustrar esse processo, a Figura 25 apresenta um exemplo em que, para uma tarefa de classificação de texto, uma rede neural de camada

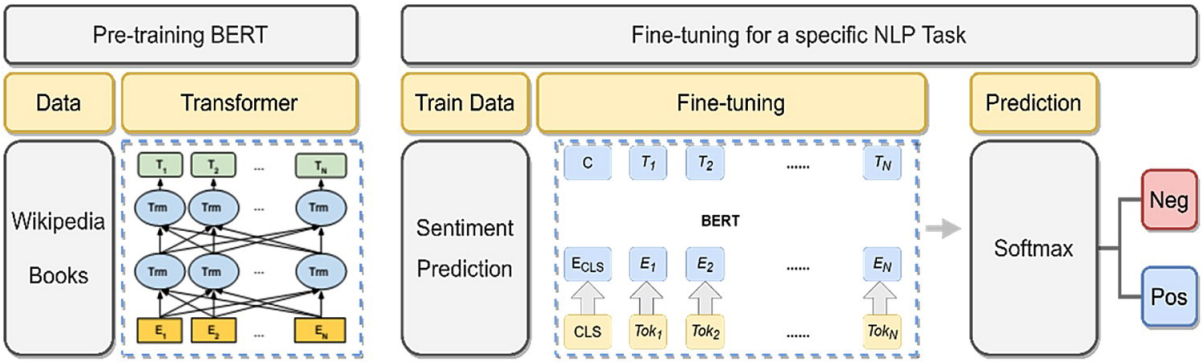


Figura 25 – Fine Tuning. Fonte: (OZCIFT et al., 2021).

única com saída *softmax* é acrescentada sobre o modelo BERT ajustado. Um exemplo de esquema de classificação binária de sentimento (negativa ou positiva) é exibido nesta Figura 25.

Durante o *fine-tuning*, as sentenças de entrada são processadas pelo *encoder* do *transformer*, utilizando os pesos pré-treinados. A saída do *encoder*, que contém as representações contextuais das palavras, é passada para a camada adicional, que produz a predição final. Utiliza-se um conjunto de dados específico para a tarefa, e a perda entre a predição e a classe real é utilizada para ajustar todos os pesos do modelo através de *backpropagation*. Assim, o modelo ajustado realiza a tarefa específica, utilizando as representações contextuais refinadas para obter um melhor desempenho.

3 Referencial Teórico sobre Modelagem de Tópicos

3.1 Definição e Aplicações da Modelagem de Tópicos

A prática de modelagem compreende a identificação e análise dos tópicos presentes em documentos sem intervenção direta na definição desses tópicos. Tal procedimento é conduzido através da investigação de um corpus de dados documentais, visando discernir e delinear os temas relevantes dentro do conjunto textual. Este processo almeja desvendar padrões e estruturas substanciais nos documentos, oferecendo perspicácias valiosas acerca do conteúdo e da estrutura organizacional do corpus sob análise (ABDELRAZEK et al., 2023).

A Modelagem de Tópicos (MT) é uma abordagem essencial para compreender e organizar grandes volumes de documentos textuais. Um dos principais objetivos dessa abordagem é a identificação de dois elementos fundamentais: coleções e tópicos (ABDELRAZEK et al., 2023). Os tópicos são constituídos de elementos básicos que se unem para formar uma coleção, frequentemente constituídas por palavras agrupadas em documentos ou conjuntos de palavras. Por sua vez, um tópico é definido como um conjunto de termos que, em conjunto, descrevem um significado semântico puro. Este conceito de tópico é concebido como uma coleção homogênea cujos termos compartilham semelhanças semânticas.

A MT tem encontrado aplicação em uma variedade de campos. As tarefas de processamento de linguagem natural (PLN), por exemplo, têm se beneficiado significativamente da modelagem de tópicos, especialmente em áreas como resumo de texto e análise de sentimentos. Além disso, tem sido amplamente explorada em outros campos, tais como bioinformática e economia.

A MT tem potencial para contribuir em diversas áreas e na exploração de dados. Um exemplo disso é a sua aplicação nas redes sociais, onde os modelos de tópicos foram empregados para compreender o nível de satisfação dos clientes em relação aos aspectos/tópicos associados aos produtos de uma empresa (JEONG; YOON; LEE, 2019). Na área de transportes, pesquisadores empregaram a modelagem de tópicos para investigar as tendências, possibilitando a identificação de padrões gerais, como sustentabilidade e comportamento de viagem, bem como outras tendências temporais (SUN; YIN, 2017).

A MT encontra aplicação na área da economia. Por exemplo, em um estudo discutido por (AMBROSINO et al., 2018), foi realizada uma investigação sobre a evolução

histórica da economia ao longo de uma janela de tempo específica, analisando sua estrutura. O objetivo era identificar os temas mais relevantes discutidos pelos economistas em cada década do século XX.

Na bioinformática, os autores conduziram uma revisão abrangente sobre a aplicação e desenvolvimento de modelagem de tópicos na área, além de analisarem o campo da bioinformática (HEO et al., 2017; LIU et al., 2016). Na engenharia de software, modelagem de tópicos têm sido utilizados em tarefas como análise de código-fonte (DIT et al., 2013), testes (HEMMATI et al., 2017), análise de requisitos (HINDLE et al., 2012), arquitetura de software (GARCIA et al., 2011).

A MT representa um campo desafiador e rico em oportunidades, cujos desafios podem ser categorizados em três grupos distintos: aprimoramento de modelos, avaliação automatizada e descoberta de dados (ABDELRAZEK et al., 2023). Em nossa abordagem para a modelagem de tópicos em redes sociais, conduziremos uma análise dos tweets provenientes dos Institutos Federais (IFs) brasileiros. Nesse sentido, a aplicação de estratégias de modelagem de tópicos eficazes podem contribuir para o aprimorarento da análise de tópicos contido no modelo. Este foco nos aspectos e tópicos é fundamental para a interpretação dos dados coletados.

Ademais, no âmbito da avaliação automatizada, nosso trabalho proposto visa a extração automática de tópicos a partir do conjunto de dados dos IFs, visando proporcionar uma abordagem eficaz para a análise de tweets. Por fim, no contexto da descoberta de dados, nosso trabalho almeja a obtenção de informações em rede sociais sobre os IFs, contribuindo para enriquecer ainda mais a análise e compreensão presentes nessas instituições de ensino.

A escolha adequada e a otimização dos hiperparâmetros dos modelos, bem como a regularização, são essenciais para alcançar um desempenho consistente e confiável, sendo atualmente um tema ativo de pesquisa. Além disso, há um crescente interesse na adaptação dos modelos às características específicas dos corpus, especialmente em relação a textos curtos ou fragmentados, o que exige estratégias de pré-processamento e configuração de hiperparâmetros cuidadosamente ajustadas (SIA; DUH, 2021).

Neste contexto, é imprescindível que nossa abordagem seja meticulosa, visando mitigar ruídos (informações irrelevantes, aleatórias ou imprecisas nos dados) e dados enviesados (aqueles com uma inclinação ou distorção em direção a determinados resultados), especialmente considerando a peculiaridade do ambiente do Twitter. Este desafio assume uma relevância significativa dada a natureza dinâmica e diversificada das interações nesta plataforma de mídia social. Além disso, para alcançar um desempenho consistente e confiável, é fundamental testar e ajustar uma variedade de hiperparâmetros nos modelos de análise. Adicionalmente, é essencial considerar a influência de fatores externos, como tendências, eventos atuais e variações temporais, os quais podem impactar consideravelmente a interpretação e análise dos dados coletados.

Outra área de pesquisa é a avaliação automática(HOYLE et al., 2021; DOOGAN; BUNTINE, 2021; KOLTCOV et al., 2021). Embora a avaliação humana seja frequentemente considerada um padrão essencial no contexto do processamento de linguagem natural, sua implementação é geralmente dispendiosa, cara e sujeita a erros. Além disso, a falta de um sistema de avaliação unificado e de *benchmarks* padronizados dificulta a comparação entre diferentes abordagens e modelos. A diversidade de processos e métricas utilizadas na avaliação torna-se um obstáculo significativo para o avanço coeso do campo. Diante desse cenário, torna-se evidente estabelecer *benchmarks* confiáveis e um sistema de avaliação unificado que possa facilitar a comparação e o desenvolvimento contínuo dos modelos de modelagem de tópicos (TERRAGNI et al., 2021).

Diante destas questões levantadas, nossa proposta visa preencher uma lacuna importante na pesquisa ao criar um *benchmark* específico e uma coleção de dados para análise de modelagem de tópicos no contexto das instituições federais, com foco no Twitter. Essa iniciativa tem o potencial de contribuir para o avanço da área ao fornecer um conjunto de dados padronizado e uma referência consistente para avaliar diferentes abordagens e modelos de modelagem de tópicos. Ao estabelecer um *benchmark* confiável e um sistema de avaliação unificado, nossa proposta busca facilitar a comparação entre diferentes métodos de Modelagem de Tópicos, bem como o desenvolvimento contínuo desses modelos. Isso é crucial, considerando a diversidade de processos e métricas utilizadas na avaliação, que têm sido um obstáculo para o progresso coeso do campo.

3.2 Técnicas de Modelagem de Tópicos

Em nossa abordagem, adotamos 3 representantes de famílias de Modelagem de Tópicos, sendo elas o LDA, que trata a modelagem de forma probabílistica, BERTopic que utiliza redes neurais com arquitetura *transformers* e NMF baseado em fatoração de matrizes.

3.2.1 Latent Dirichlet Allocation (LDA)

O modelo de *Latent Dirichlet Allocation* (LDA) introduzido por (BLEI; NG; JORDAN, 2003), é uma técnica amplamente utilizada na análise de tópicos em conjuntos de documentos textuais. A premissa fundamental do LDA é que cada documento pode ser visto como uma combinação de tópicos, onde cada tópico é uma distribuição de palavras. Antes de adentrar nos detalhes do modelo LDA, será examinado superficialmente as palavras que compõem o nome da técnica.

O termo *Latent* indica que o modelo descobre tópicos "ainda não encontrados" ou "ocultos" nos documentos. *Dirichlet* denota a suposição do LDA de que tanto a distribuição de tópicos em um documento quanto a distribuição de palavras nos tópicos seguem

distribuições de *Dirichlet*. No processo generativo do LDA, o resultado da distribuição de *Alocattion* é utilizado para atribuir as palavras do documento a diferentes tópicos (BLEI, 2012).

Operando por meio de uma abordagem probabilística, o LDA (Latent Dirichlet Allocation) atribui as palavras em um documento a tópicos específicos com base em sua probabilidade de ocorrência (GRIFFITHS; STEYVERS, 2004). Essa modelagem probabilística permite uma análise automatizada e precisa dos principais temas presentes nos dados textuais, facilitando a compreensão e organização de grandes volumes de documentos.

Para ilustrar a ideia desta técnica, na Figura 26, são apresentadas as intuições fundamentais do (LDA). No contexto deste modelo, presume-se que uma variedade de "tópicos", os quais representam palavras, esteja presente em toda a coleção (à esquerda). Cada documento é gerado da seguinte maneira: primeiro, é selecionada uma distribuição sobre os tópicos (o histograma à direita); em seguida, para cada palavra, é feita uma atribuição de tópico (representada pelos marcadores coloridos) e a palavra é selecionada do tópico correspondente (BLEI, 2012).

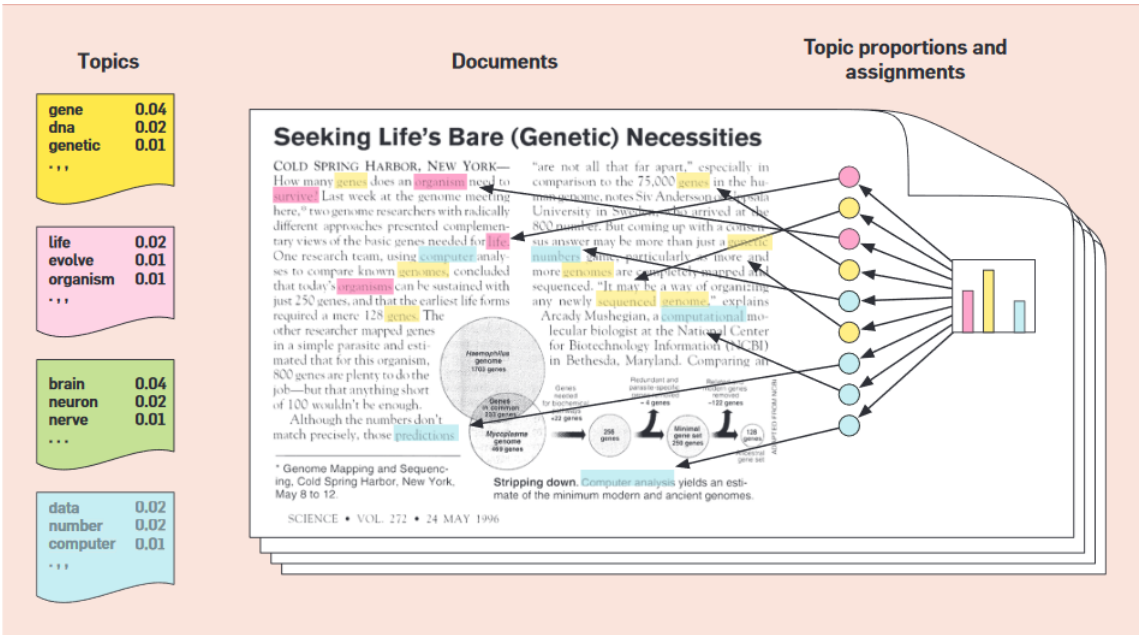


Figura 26 – Intuição do LDA. Fonte: (BLEI, 2012)

Pode-se observar na figura 26 que termos relacionados à análise de dados, como "computador" e "predição", são destacados em azul; palavras pertinentes à biologia evolucionária, como "vida" e "organismo", são representadas em rosa; e termos associados à genética, como "sequenciado" e "genes", são realçados em amarelo. Se cada palavra do artigo fosse realçada, seria evidente que o texto incorpora conceitos de genética, análise de dados e biologia evolutiva em diferentes proporções. A distribuição de tópicos no texto reflete a probabilidade associada a cada um desses temas, e cada palavra é atribuída a um

desses três tópicos (BLEI, 2012).

Essa intuição é formalizada e representada pelo modelo gráfico do LDA (Latent Dirichlet Allocation), conforme ilustrado na Figura 27.

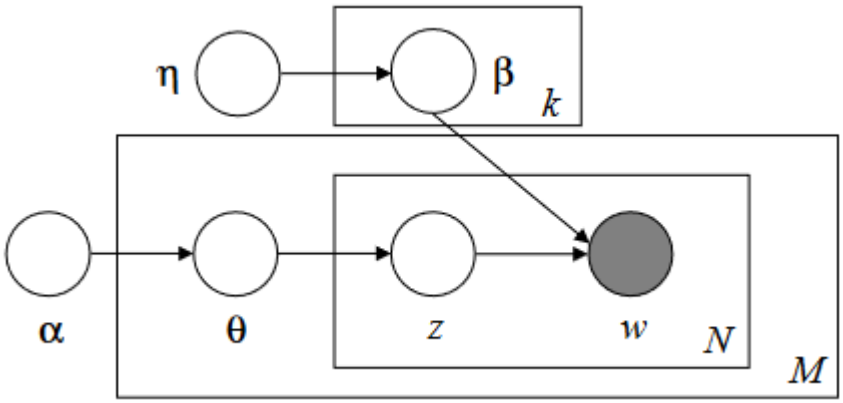


Figura 27 – Modelo gráfico do LDA. Fonte: (BLEI; NG; JORDAN, 2003)

As notações utilizadas nesta figura, conforme descritas por (BLEI; NG; JORDAN, 2003), são as seguintes:

- α: O hiperparâmetro da distribuição de *Dirichlet*, relacionado à distribuição documento-termo. Este parâmetro controla a dispersão da distribuição de tópicos em um documento.
- θ: A distribuição de tópicos para um documento específico. Para cada documento d, θ_d é a distribuição sobre os tópicos.
- β: O hiperparâmetro da distribuição de *Dirichlet*, relacionado à distribuição tópico-palavra. Este parâmetro controla a dispersão da distribuição de palavras em um tópico.
- η: Hiperparâmetro para a distribuição de tópicos.
- z: A atribuição de tópico para cada palavra em um documento. Para cada palavra w, z é o tópico atribuído a essa palavra.
- w: Uma palavra específica em um documento.
- N: O número de palavras no vocabulário.
- M: O número total de documentos.
- k: O número de tópicos no modelo.

Para ilustrar o modelo, a equação 3.1 descreve a probabilidade no modelo LDA. Os parâmetros α e β atuam como priors de *Dirichlet* (BLEI; NG; JORDAN, 2003), agindo como hiperparâmetros do modelo. Cada tópico k é modelado como uma distribuição de *Dirichlet* $\beta_K \sim Dir(\eta)$ dentro de um corpus. O corpus representa o conjunto de todas as palavras contidas nos documentos analisados. Cada documento d no corpus possui uma distribuição de tópicos representada por $\theta_d \sim Dir(\alpha)$. Além disso, cada palavra individual em um documento segue uma distribuição multinomial condicionada pela distribuição de tópicos do documento, $Z_{d,n} \sim Mult(\theta_d)$, e pela distribuição de palavras associada ao tópico $z_{d,n}$, $W_{d,n} \sim Mult(\beta_{z_{d,n}})$. A distribuição de *Dirichlet* implica que as distribuições a posteriori e a priori seguem a mesma distribuição, simplificando assim a inferência do LDA.

$$p(\beta, \theta, z, w) = \left(\prod_{k=1}^{K} p(\beta_k|\eta)\right)\left(\prod_{d=1}^{D} p(\theta_d|\alpha) \prod_{n=1}^{N} p(z_{d,n}|\theta_d)p(w_{d,n}|\beta_{1:K}, z_{d,n})\right) \tag{3.1}$$

Podemos associar o LDA com o algoritmo 1. Primeiramente, escolhe-se aleatoriamente uma distribuição sobre os tópicos. Para cada tópico k, uma distribuição Dirichlet β_k é amostrada. Em seguida, no início do segundo loop, para cada documento, é escolhida aleatoriamente uma distribuição de tópicos representada por $\theta_d \sim Dir(\alpha)$.

Em sequência, para cada palavra no documento, escolhe-se aleatoriamente um tópico a partir da distribuição, no Algoritmo 1, isso é representado no *loop* como $Z_{d,n} \sim Mult(\theta_d)$. Em seguida, escolhe-se aleatoriamente uma palavra a partir do tópico (distribuição sobre as palavras), representada no loop como $W_{d,n} \sim Mult(\beta_{z_{d,n}})$.

Algorithm 1 Pseudocódigo do LDA

Dados:
K = número de tópicos na coleção dos documentos;
D = coleção de documentos;
N = número de termos nos documentos;
Parâmetros:
α: distribuição de tópicos por documento;
η: distribuição de termos por tópico;

for $k = 1$ to K **do**
  $\beta_k \sim Dir(\eta)$;
end for
for $d = 1$ to D **do**
  $\theta_d \sim Dir(\alpha)$;
  for $n = 1$ to N **do**
    $Z_{d,n} \sim Mult(\theta_d)$;
    $W_{d,n} \sim Mult(\beta_{z_{d,n}})$;
  end for
end for

De acordo com (BLEI; NG; JORDAN, 2003), na representação gráfica do modelo LDA na figura 27, os retângulos são designados como "placas", indicando a replicação. O quadro exterior representa os documentos, enquanto o quadro interior retrata a repetida seleção de tópicos e palavras dentro de um documento.

Esse modelo apresenta uma estrutura em três níveis distintos de variáveis. Os parâmetros α e β são considerados parâmetros de nível de coleção, presumindo-se que são amostrados uma única vez durante o processo de construção de um conjunto de documentos. Em seguida, as variáveis θ representam o nível de documento, sendo amostradas individualmente para cada documento. Por fim, as variáveis z e w representam o nível de palavra, sendo amostradas para cada palavra em cada documento. Essa estrutura permite modelar a distribuição de tópicos em documentos e atribuir palavras a esses tópicos, oferecendo uma maneira de compreender a estrutura fundamental de grandes conjuntos de documentos (BLEI; NG; JORDAN, 2003).

No que diz respeito aos parâmetros de nível de coleção, os hiperparâmetros α e β se destacam. O hiperparâmetro α controla a distribuição de tópicos por documento. Valores menores de α tendem a resultar em documentos com menos tópicos, enquanto valores mais altos permitem uma maior diversidade de tópicos em um documento. Por outro lado, valores menores de β resultam em tópicos mais concentrados em um conjunto restrito de palavras, enquanto valores mais altos de β permitem uma maior variabilidade de palavras em um tópico.

3.2.2 Non-Negative Matrix Factorization (NMF)

Non-negative matrix factorization (NMF) é uma técnica de álgebra linear aplicada no campo do aprendizado de máquina não supervisionado, destaca-se como uma abordagem eficaz para identificar tópicos em conjuntos de dados. A NMF, introduzida por (LEE; SEUNG, 1999), oferece uma abordagem poderosa e intuitiva para a análise de dados e a extração de padrões de informação. Esses algoritmos têm a capacidade de extrair automaticamente características significativas de um conjunto de vetores de dados não-negativos.

De maneira geral, dado uma matriz não negativa de A que representam os termos em cada documento. O objetivo é encontrar duas outras matrizes W e H, que representam, respectivamente, os termos contidos em cada tópico e os tópicos contidos em cada documento, conforme descrito por (LEE; SEUNG, 2001), este processo é representado pela seguinte equação:

$$A \approx WH \tag{3.2}$$

De maneira intuitiva, podemos representar na Figura 28, onde ilustra o NMF,

primeiramente de uma matriz, representado por A, consistindo de m palavras em n documentos. Em seguida, em uma matriz representada por W não negativa das m palavras originais por k tópicos. E por fim, uma matriz H, com esses mesmos k tópicos pelos n documentos originais.

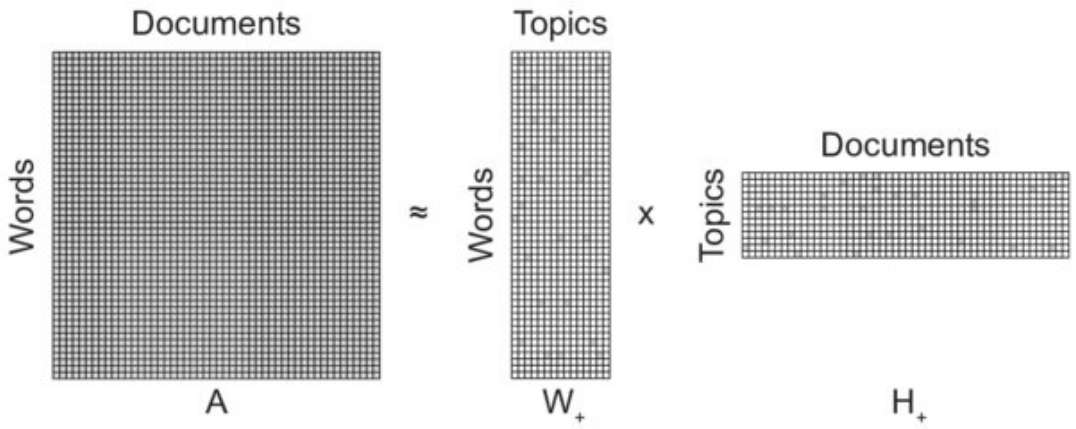


Figura 28 – Non-negative matrix factorization. Fonte: (KUANG; BRANTINGHAM; BERTOZZI, 2017)

.

Para melhor entendimento, a equação descrita anteriormente 3.2, pode ser vista da seguinte forma:

$$A_{m\times n} = W_{m\times k} H_{k\times n} \tag{3.3}$$

Tais notações, podem ser descritas da seguinte maneira:

- A: Matriz de entrada original (Combinação linear de W e H).
- W: A matriz base contendo vetores ou componentes não negativos.
- H: Matriz de Coeficientes (Pesos associados com w).
- k: O número de componentes (ou vetores base) escolhidos para a factorização.
- m: Número de exemplos no conjunto.
- n: Número de documentos.

No contexto da análise estatística de dados multivariados, a aplicação da NMF segue um processo específico, conforme visto na equação 3.3: inicialmente, é fornecido um conjunto de vetores de dados multivariados de n dimensões. Esses vetores são organizados nas colunas de uma matriz A de dimensões $m \times n$, onde m é o número de exemplos no conjunto de dados.

Em seguida, a matriz A é decomposta aproximadamente em duas matrizes: Matriz W de dimensões $m \times k$ e uma matriz H com dimensões $k \times n$. Normalmente, k é escolhido para ser menor do que n ou m, de modo que W e H tenham dimensões menores que a matriz original A. Esse procedimento resulta em uma versão comprimida da matriz de dados original. (LEE; SEUNG, 2001)

3.2.3 BERTopic

O BERTopic é uma estratégia de modelagem de tópicos que utiliza uma abordagem de modelagem para identificar tópicos ocultos em uma coleção de documentos. Ele aprimora esse processo extraindo representações coesas de tópicos por meio de uma variação do TF-IDF baseada em classe (GROOTENDORST, 2022).

Modelos convencionais de Modelagem de Tópicos, como LDA e NMF, tratam documentos como *bags-of-words*, uma representação simplificada que foca nas palavras individuais e suas frequências nos documentos, modelando cada documento como uma combinação de tópicos ocultos. No entanto, esses modelos têm a limitação de ignorar as relações semânticas entre as palavras, devido à sua abordagem de *bag-of-words*. Isso significa que a estrutura sintática e o contexto das palavras nas sentenças podem ser desconsiderados, o que pode resultar em representações imprecisas dos documentos (GROOTENDORST, 2022).

Para superar essa limitação, técnicas avançadas de *embeddings* textuais, como o BERT (DEVLIN et al., 2019), tornaram-se populares no campo de processamento de linguagem natural. Essas técnicas capturam relações semânticas complexas entre palavras e sentenças, gerando representações vetoriais que refletem o significado contextual dos textos. Assim, o BERTopic aproveitam as vantagens dos *embeddings* contextuais para proporcionar representações mais precisas e semânticas dos tópicos discutidos nos documentos, oferecendo uma abordagem mais robusta e eficaz na análise e categorização de textos.

O BERTopic opera em três fases distintas. Primeiramente, cada documento é convertido em um *embedding* usando um modelo de linguagem pré-treinado, como o BERT. Em seguida, a dimensionalidade desses *embeddings* é reduzida para melhorar a clusterização. Por fim, os clusters de documentos são analisados, e as representações de tópicos são extraídas utilizando uma adaptação customizada do TF-IDF baseada em classes (c-TF-IDF) (GROOTENDORST, 2022).

Para uma compreensão detalhada do funcionamento do BERTopic em cada uma dessas etapas, exploraremos nas próximas subseções.

3.2.3.1 Sentence Embeddings

Nesta etapa, o BERTopic gera *embeddings* de documentos para criar representações vetoriais, assumindo que documentos sobre o mesmo tópico são semanticamente similares. O BERTopic utiliza o framework *Sentence-BERT* (SBERT) (REIMERS; GUREVYCH, 2019), para essa etapa, que converte sentenças e parágrafos em representações densas de vetores usando modelos de linguagem pré-treinados.

O *Sentence-BERT* (SBERT) é uma variação da rede BERT que utiliza arquiteturas de redes siamesas e de trios, possibilitando a criação de *embeddings* de sentenças com significados semânticos. Isso permite que o BERT seja aplicado a novas tarefas que anteriormente não obtinham resultados satisfatórios, como comparação de similaridade semântica em grande escala, agrupamento (clustering) e recuperação de informações através de busca semântica (REIMERS; GUREVYCH, 2019).

O BERT foi é capaz de obter resultados satisfatórios em tarefas de classificação de sentenças e regressão de pares de sentenças. Ele utiliza um *encoder* onde duas sentenças são processadas simultaneamente pela rede *transformer*, e o valor alvo é previsto. Essa abordagem resulta em cálculos de similaridade semântica entre todas as combinações possíveis de sentenças. Logo, essa combinação que é quadrática em relação ao número de sentenças, podem ser consideradas computacionalmente caro (REIMERS; GUREVYCH, 2019).

Por outro lado, o SBERT foi desenvolvido para criar embeddings de sentenças semanticamente significativos e eficientes para tarefas como comparação de similaridade, clustering e busca semântica. O SBERT adota uma arquitetura de rede siamesa/tripla que ajusta o BERT para gerar vetores de tamanho fixo para cada sentença individualmente. Esses vetores podem ser comparados usando medidas de similaridade, como similaridade cosseno, de maneira muito mais eficiente. (REIMERS; GUREVYCH, 2019).

3.2.3.2 Redução da dimensionalidade e Clusterização

A clusterização, que organiza dados semelhantes em grupos (*clusters*), é aprimorada pela redução da dimensionalidade, que simplifica os dados mantendo suas características essenciais. Entre as várias técnicas de clusterização, a redução da dimensionalidade dos embeddings é mais direta. O UMAP (*Uniform Manifold Approximation and Projection for Dimension Reduction*) provou ser eficaz na preservação das características locais e globais dos dados de alta dimensionalidade em dimensões reduzidas (MCINNES; HEALY; MELVILLE, 2018). Além disso, sem restrições computacionais quanto às dimensões dos embeddings, o UMAP pode ser aplicado a modelos de linguagem com diferentes espaços dimensionais. Portanto, o UMAP é utilizado no BERTopic para reduzir a dimensionalidade dos embeddings dos documentos gerados.

Os *embeddings* reduzidos são agrupados utilizando o HDBSCAN (*Hierarchical DBSCAN*), um algoritmo de clusterização baseado em densidade hierárquica derivado do DBSCAN (*Density-Based Spatial Clustering of Applications with Noise*). O HDBSCAN modela clusters usando uma abordagem de clusterização suave, permitindo que ruídos (dados indesejados, irrelevantes ou distorcidos) sejam tratados como outliers (dados atípicos ou incomuns). Essa abordagem é essencial para evitar que dados irrelevantes ou distorcidos prejudiquem a formação dos clusters, melhorando assim as representações de tópicos. O HDBSCAN organiza os dados em uma hierarquia de clusters com base na densidade dos pontos. Esse método utiliza a ideia de densidade para identificar clusters, focando nas regiões onde os dados estão mais próximos uns dos outros, ou seja, nas áreas de maior densidade (CAMPELLO; MOULAVI; SANDER, 2013).

De maneira geral, os clusters são gerados a partir do agrupamento dos embeddings dos tópicos. Cada embedding de tópico, por sua vez, é gerado com a média ponderada dos embeddings de termos dos tópico com base em seus valores c-TF-IDF. Portanto, em última instância, a representação vetorial dos termos é agrupada em tópicos com a aplicação de algoritmo de clusterização hierárquico, como o HDBSCAN. Cada documento é inicialmente dividido em sentenças e para cada sentença são gerados embeddings que capturam a semântica contextual. Esses embeddings são então combinados em um espaço vetorial multidimensional, onde o HDBSCAN identifica regiões de alta densidade para formar clusters de embeddings similares. O algoritmo constrói uma hierarquia de clusters (tópicos), onde pontos que não se encaixam em regiões densas são considerados ruído. Essa hierarquia é representada por uma árvore de clusterização que pode ser condensada para revelar estruturas de clusters em diferentes níveis de granularidade.

3.2.3.3 Representação de tópicos

Uma vez que os documentos estão agrupados em clusters e a dimensionalidade foi reduzida com UMAP, podemos então calcular uma distribuição de palavras para cada cluster. As representações de tópicos são fundamentadas nos documentos de cada cluster, onde cada cluster é atribuído a um tópico. Podemos aprimorar o TF-IDF, uma métrica de importância de palavras em documentos, para destacar a relevância de termos para cada tópico. O TF-IDF tradicional combina a frequência dos termos com a frequência inversa dos documentos (JOACHIMS, 1996), como expresso na equação:

$$W_{t,d} = tf_{t,d} \cdot \log\left(\frac{N}{df_t}\right) \tag{3.4}$$

Nesta equação 3.4, $tf_{t,d}$ representa a frequência do termo t no documento d, enquanto df_t é o número de documentos em um corpus que contêm o termo t, e N é o total de documentos no corpus.

Expandindo esse conceito para clusters de documentos, primeiro agrupamos todos os documentos de um cluster em um único documento, agregando-os. Em seguida, ajustamos o TF-IDF para essa representação, transformando documentos em clusters:

$$W_{t,c} = tf_{t,c} \cdot \log\left(1 + \frac{A}{tf_t}\right) \tag{3.5}$$

Nesta equação 3.5, o $tf_{t,c}$ denota a frequência do termo t em um cluster c. A classe c representa a coleção de documentos concatenados em um único documento para cada cluster. A frequência inversa dos documentos é substituída pela frequência inversa da classe para medir a importância de um termo para uma classe. Isso é calculado pelo logaritmo da média de palavras por classe A dividido pela frequência do termo t em todas as classes. Para manter apenas valores positivos, adicionamos um à divisão dentro do logaritmo (GROOTENDORST, 2022).

Assim, este método de TF-IDF baseado em classe modela a importância das palavras em clusters em vez de documentos individuais. Isso possibilita a geração de distribuições de palavras por tópico para cada cluster de documentos. Por fim, ao mesclar de forma iterativa as representações c-TF-IDF do tópico menos comum com o mais similar, podemos reduzir o número de tópicos para um valor definido pelo usuário (GROOTENDORST, 2022).

3.3 Modelagem de Tópicos em Redes Sociais

As redes sociais são dinâmicas, e variam de acordo com normas sociais, comportamento do consumidor, práticas jornalísticas e estratégias organizacionais de mídia. De fato, pesquisas recentes sobre modelagem de tópicos para textos curtos tem se concentrado nos desafios apresentados pelos conjuntos de dados das redes sociais, comumente avaliados por medidas automatizadas que não garantem a produção de tópicos significativos. A falta de entendimento sobre como os pesquisadores utilizam esses modelos, juntamente com possíveis problemas na construção e aplicação dos mesmos, ameaça a validade do desenvolvimento e dos insights gerados (LAUREATE; BUNTINE; LINGER, 2023).

Diversas plataformas de mídia social impõem restrições ao número de caracteres em cada postagem, como é o caso do Twitter, que limita cada postagem a um máximo de 280 caracteres (QUEIROZ, 2018). No entanto, os textos nessas plataformas digitais são frequentemente compostos por palavras não padronizadas, repletas de dialetos, abreviações, negligência gramatical e erros ortográficos (ARIFFIN; TIUN, 2020). Além disso, os textos em redes sociais podem estar desestruturados e incluir elementos como hashtags, emojis e gírias, o que dificulta a compreensão e resulta em uma complexidade textual nas publicações de texto curto (EGGER; YU, 2021). Desta forma, essa variedade de texto torna-se difícil

de processar devido ao alto nível de ruído e à estrutura textual distinta.

Para lidar com dados desestruturados, é essencial realizar o pré-processamento dos dados de texto. Diversas estratégias são empregadas para essa finalidade. Por exemplo, podem ser realizados a remoção de números e sinais pouco informativos, os quais são excluídos dos dados de texto. Adicionalmente, é utilizada a estratégia de remoção de *stopwords*, que são palavras que não possuem significado especial podem não contribuir para a extração de tópicos dos textos. Além disso, é utilizado o método de *stemming* para tratar palavras com significados idênticos, mas que são percebidas como diferentes devido a variações nas terminações das palavras causadas por mudanças no tempo verbal ou número (singular/plural). Esse método extrai apenas as raízes das palavras, permitindo que palavras semelhantes sejam tratadas como idênticas. (HONG; PARK; CHOI, 2020).

Apesar das dificuldades enfrentadas, os métodos de modelagem de tópicos são aplicados nas redes sociais de diversas maneiras. Esses métodos incluem técnicas como a alocação latente de Dirichlet (LDA), fatorização de matriz não negativa (NMF) e BERTopic. Os resultados obtidos incluem informações sobre os principais tópicos discutidos, a detecção de tendências e mudanças ao longo do tempo, e a compreensão das opiniões e sentimentos dos usuários sobre determinados assuntos. Essa análise ajuda a entender melhor as interações e discussões na plataforma do Twitter, contribuindo para uma compreensão mais ampla das conversas online e suas implicações em diferentes áreas, como saúde (PINTO et al., 2020), política (CARMO et al., 2023; HOANG; COHEN; LIM, 2014), análise de sentimentos (XIANG; ZHOU, 2014; ALAMSYAH et al., 2018).

No contexto da saúde, mais especificamente, uma pesquisa conduzida por (PINTO et al., 2020), utilizou a modelagem de tópicos em dados do Twitter para analisar as discussões sobre a pandemia de COVID-19. Essa abordagem não apenas identificou os tópicos mais discutidos, mas também investigou a relação entre esses tópicos e os sentimentos dos usuários, fornecendo uma análise detalhada da dinâmica das discussões online durante a pandemia.

Neste contexto da pandemia, a análise de tópicos em dados do Twitter revelou uma variedade de discussões, alguns tópicos geraram repercussões exclusivamente negativas, como o pânico relacionado à necessidade de estocar alimentos *(Food)* e o aumento de golpes pela internet *(Scams)*. Por outro lado, foram identificados tópicos que refletiram sentimentos positivos, como a conscientização sobre a propagação do vírus *(Spread)* e o isolamento social *(Quarantine e Home)*. Além disso, alguns tópicos foram modelados com sentimentos tanto negativos quanto positivos, como as discussões sobre compras *(Shopping)*, supermercados *(Supermarket)* e saúde *(Health)*.

No campo político, a modelagem de tópicos foi empregada para analisar tweets, com ênfase nos dados de respostas diretas às mulheres na política durante o primeiro turno das eleições de 2022, incluindo candidatas ao Senado ou ex-senadoras (CARMO et al.,

2023). Durante este estudo, observou-se um tópico com maior ênfase no público brasileiro, abordando temas "brasileiros", inclusive com menções à palavra "presidente", remetendo ao contexto eleitoral. Como esperado, foi identificado um tópico específico dedicado às mulheres. Por fim, também se destacou um tópico relacionado ao Senado, possivelmente associado à participação de uma das mulheres na política.

Outro exemplo de aplicação da modelagem de tópicos no campo político foi apresentado por (HOANG; COHEN; LIM, 2014). Neste estudo, foram analisados dois conjuntos de dados de tweets: o primeiro consistia nos tweets de senadores do 112º Congresso dos EUA, no período entre maio de 2012 e fevereiro de 2013; o segundo conjunto abrangia os tweets de 56 usuários políticos populares antes das eleições presidenciais dos EUA de 2012.

Nesse exemplo, a modelagem de tópicos foi utilizada para identificar comunidades de usuários em redes sociais, levando em consideração os sentimentos expressos e os comportamentos adotados. Na análise, os tópicos associados ao perfil de cada comunidade são representativas de suas respectivas ideologias políticas: "liberal", "progressista", "democratas", para a comunidade Democrata; "conservador", "cristão", para a comunidade Republicana; e "mídia", "esporte", "música", "editor", para a comunidade Neutra, que inclui principalmente contas de pessoas e associações profissionais.

Na análise de sentimentos, em um estudo conduzido por (ALAMSYAH et al., 2018), foram analisadas as opiniões dos consumidores sobre a empresa Uber, uma das líderes mundiais em transporte, utilizando como fonte de dados a plataforma Twitter. O objetivo foi realizar uma análise de conteúdo, empregando modelagem baseada em tópicos para avaliar o sentimento dos temas discutidos. A pesquisa teve como propósito mapear a opinião pública e analisar o sentimento expresso nos textos dos consumidores.

Nesta análise, os principais tópicos positivos nos dados do Uber são sobre promoções e elogios aos motoristas, indicando uma reação positiva dos usuários em relação aos serviços. No entanto, também foram identificados tópicos negativos, sendo o mais proeminente sobre casos de assédio, seguido por reclamações sobre o comportamento dos motoristas, incluindo relatos de embriaguez ao volante e conversas excessivas, mesmo quando os clientes preferem silêncio.

Em outro estudo, conforme referido por (XIANG; ZHOU, 2014), foram utilizados dois conjuntos de tweets: o primeiro conjunto consiste em 2 milhões de tweets selecionados aleatoriamente da coleção de janeiro e fevereiro de 2014. O outro conjunto é composto por 74 mil documentos de notícias, totalizando 50 milhões de palavras, coletados durante o primeiro semestre de 2013 de agências de notícias online. Nesse contexto, foi empregada a abordagem de sentimentos baseados em tópicos, juntamente com a análise dos tópicos mais frequentes para compreender as tendências e os padrões de sentimentos expressos nas mensagens. Foram fornecidos alguns exemplos dos tópicos identificados tanto em tweets quanto nos dados de agências de notícias. As palavras mais frequentes em cada

tópico foram listadas. E os tópicos abordaram sobre telefones, esportes, vendas e política, respectivamente.

3.4 Modelagem de Tópicos em Contextos Educacionais

A modelagem de tópicos é uma abordagem amplamente utilizada em instituições de ensino, tanto para compreender as experiências dos estudantes no ensino superior (HONG; PARK; CHOI, 2020) quanto para a análise de crises universitárias (SANANDRES; ABELLO; MADARIAGA, 2020). Além disso, é aplicada em sistemas virtuais de aprendizado (KUANG; LUO; SUN, 2011).

A modelagem de tópicos proporciona uma abordagem eficaz para identificar as tendências de investigação na experiência dos estudantes no ensino superior. Essa metodologia permite analisar os temas abordados, fornecendo uma compreensão da evolução dos campos de estudo relacionados à experiência do estudante e das principais tendências ao longo do tempo. (HONG; PARK; CHOI, 2020).

Usando o método LDA, foram identificados 21 tópicos relevantes. As probabilidades foram calculadas para cada termo dentro de cada tópico específico. Alguns exemplos dos tópicos podem ser mencionados. Como por exemplo, para o tópico 1, a probabilidade de aparecimento foram altas para os termos “ensino superior” e “social”, com uma probabilidade de ocorrência de um termo dentro do tópico de 18,5% e 12,7%, respectivamente, seguidos por “participar” e “ampliar”, com 6,4% e 4,4%, respectivamente. Este tópico foi denominado de Ampliar a participação no ensino superior.

O Tópico 2, intitulado de Transições no primeiro ano, apresentou “universidade” como o termo principal, com 19,9% de probabilidade, seguido por “transição(ões)” e “primeiro ano”, com 9,9% e 8,6%, respectivamente. No Tópico 4, denominado de Percepções dos alunos, “experience(s)” e “student(s)” representaram 22,3% e 20,2%, respectivamente, seguidos por “percept(s)” com 8,1%. O Tópico 6, intitulado como Educação profissional, destacou “educação” com 32,5% de probabilidade, seguido por “enfermeira(s)”, “professora(s)” e “clínica(s)”. O Tópico 7, referente a Pesquisa de ensino, revelou “ensinar” com 20,2% de probabilidade, seguido por “pesquisa” e “graduação”. O Tópico 8, denominado Avaliação e feedback, evidenciou “avaliar” e “feedback” como os termos principais, com 16,0% e 10,2%, respectivamente.

Os resultados deste estudo ressaltam a importância de avaliar as experiências dos alunos, como por exemplo, o destaque dado ao tópico 1, “Ampliar a participação no ensino superior”, cujos termos principais são “ensino superior” e “social”, com uma probabilidade de ocorrência de 18,5% e 12,7% nos documentos do tópico 1, respectivamente, sublinha a necessidade de promover políticas educacionais que visem aumentar o acesso e a inclusão no ensino superior. Da mesma forma, o tópico 2, “Transições no primeiro

ano", realça a importância de apoiar os alunos durante esse período crítico de adaptação à universidade, conforme indicado pela alta probabilidade do termo "universidade" de 19,9%. A ocorrência do tópico 4 relacionado a percepção dos alunos com a prepoderância de 22,3% nos documentos indicam a importância de avaliar as experiências dos alunos para melhorar a qualidade do ensino e garantir sua satisfação acadêmica.

Para garantir que estejam alinhadas aos objetivos institucionais e para avaliar a qualidade do suporte oferecido pelo sistema educacional. A modelagem de tópicos emerge como uma ferramenta crucial nesse processo de avaliação e compreensão das experiências dos estudantes no ensino superior (HONG; PARK; CHOI, 2020).

A modelagem de tópicos tem sido amplamente aplicada em instituições de ensino, particularmente em sistemas virtuais de aprendizado. Por exemplo, em um estudo conduzido por Kuang, Luo e Sun (2011) foi proposto um sistema de recomendação baseado na modelagem de tópicos em um ambiente virtual de ensino. As autores propuseram um modelo de interesse baseado em LDA dentro de um framework de modelagem de linguagem e o avaliaram em um sistema de *e-learning*. Esse método tem como intuito utilizar as consultas e o uso da plataforma para construir um modelo pra recomendação de materiais de estudo para os alunos. Esse método apresentou resultados 50% superiores em termos de taxas de cliques(*click-through rates*) em relação aos métodos anteriores.

Neste contexto, foi desenvolvido um sistema de recomendação com base em aproximadamente 30.000 materiais de estudo, abrangendo seis disciplinas da área da Computação. Cada disciplina foi tratada como um domínio separado, e todos os termos e elementos de conhecimento foram marcados nos materiais de estudo de cada matéria. Para proporcionar uma recomendação personalizada aos usuários, utilizou-se a modelagem de tópicos LDA. Os documentos foram mapeados para um espaço de tópicos usando um modelo de tópicos baseado em LDA. A similaridade entre os documentos foi calculada no espaço de tópicos. Os 10 documentos mais semelhantes a cada documento foram identificados e utilizados para recomendar materiais relevantes aos usuários com base em seus interesses específicos em cada disciplina.

Desta forma, neste estudo, a modelagem de tópicos, desempenhou um papel crucial na personalização da experiência educacional em plataformas online. Ao mapear documentos para um espaço de tópicos e calcular a similaridade entre eles, o sistema pode oferecer recomendações precisas e relevantes aos usuários. Essa abordagem contribui significativamente para a eficácia do aprendizado e proporcionando uma experiência mais enriquecedora aos alunos (KUANG; LUO; SUN, 2011).

Em outra vertente, pesquisadores conduziram uma análise sobre a crise enfrentada pela Universidade Nacional da Colômbia, especialmente ao examiná-la através do Twitter (SANANDRES; ABELLO; MADARIAGA, 2020). Os autores empregaram a modelagem de tópicos para investigar esse contexto. Essa metodologia se destacou como um tema

relevante na pesquisa, fornecendo informações significativas sobre as percepções e ações da universidade diante da crise.

Como resultado, foram identificados 12 tópicos relevantes, agrupados em quatro conjuntos principais. No primeiro conjunto, relacionado ao protesto social, os tópicos 1 e 3 abordam os protesto dos trabalhadores da educação, destacando aspectos como *sintraunal* (o sindicato que abrange todos os trabalhadores das universidades públicas), *protest, strike, campus, riot*, além de aspectos negativos como *confrontation, death, bombs*.

Os tópicos 2 e 4 tratam dos protestos no setor agrícola, estes tópicos indicam a presença de tweets que evidenciam a mobilização das pessoas para participar de protestos. O tópico 10, que trata dos protestos da classe trabalhadora e da intimidação dos manifestantes contendo palavras como *intimidation, blocked, abandoned, public, eviction, worker, protest,* enquanto, o Tópico 12 cobre a causa revolucionária do protesto social e inclui as palavras como *revolutionary, victory, popular, campus, strike* .

No segundo conjunto, relacionado à reforma educacional, os tópicos 5, 8 e 6 refletem a rejeição das universidades públicas à proposta de reforma, com destaque para termos como *solidarity, no to the reform, justice, march, respect, propose, threat, oblivion, save, listen, sciences, confrontation, media, classrooms, abandoned, mobilization*. Os tópicos 7 e 9, no terceiro conjunto, abordam as demandas de investimento para enfrentar a crise na universidade, enfocando melhorias na infraestrutura e na obtenção de recursos financeiros.

Por fim, o tópico 12 integra a crise na Universidade Nacional da Colômbia em um contexto mais amplo de preocupação nacional associada ao processo de paz colombiano, destacando termos como *peace, process, mobilization, research, studying, participation, talks, intellectuals, solidarity, civil*.

A modelagem de tópicos foi fundamental para compreender a complexidade do cenário educacional colombiano. A análise dos tópicos identificados revela diversos aspectos cruciais relacionados à educação. Os primeiros tópicos destacam os protestos na área da educação (Tópicos 1, 3, 10 e 12), evidenciando as reivindicações dos trabalhadores por melhores condições de trabalho e a mobilização em busca de mudanças. Em seguida, outros tópicos refletem a oposição das universidades públicas à reforma educacional, ressaltando a importância do investimento na infraestrutura (Tópicos 5, 6 e 8). Por fim, são abordadas as demandas por investimento na educação e a crise universitária, destacando a necessidade de melhorias na infraestrutura e obtenção de recursos financeiros (Tópicos 7 e 9). Essa abordagem foram essenciais para identificar o cenário educacional colombiano, discutindo questões relacionadas a protestos, reforma educacional, investimento na educação e o papel da educação no contexto nacional.

Este estudo destaca várias semelhanças com o presente trabalho de conclusão de curso, especialmente no que concerne à análise de dados textuais relacionados às instituições

de ensino. Inspirado por esse trabalho, nosso projeto visa empregar a modelagem de tópicos como uma ferramenta essencial na identificação dos temas mais relevantes na área da educação expressos nas redes sociais, utilizando o Twitter como fonte de dados. Assim como no estudo sobre a crise enfrentada pela Universidade Nacional da Colômbia, onde os principais tópicos de discussão foram identificados para compreender as preocupações, interesses e sentimentos do público relacionados àquela instituição, a modelagem de tópicos é capaz de detectar questões emergentes.

Com base nisso, ao invés de analisar a crise da Universidade Nacional da Colômbia, nosso objetivo é mais amplo e visa analisar os textos do Twitter para identificar os principais temas relacionados aos Institutos Federais (IFs) e abordar essas questões. Isso ressalta a importância da modelagem de tópicos no contexto educacional, não apenas para avaliar a opinião pública, mas também para compreender os sentimentos expressos pela comunidade acadêmica, contribuindo assim para uma compreensão mais abrangente do cenário educacional e para o aprimoramento das políticas e práticas institucionais.

3.5 Análise de Sentimentos no Contexto de Inteligência Artificial

A Análise de Sentimentos é uma subárea da Inteligência Artificial e do Processamento de Linguagem Natural (PLN) voltada para identificar automaticamente opiniões, emoções e avaliações expressas em textos. O objetivo é classificar mensagens em categorias como positivas, negativas ou neutras, permitindo compreender a percepção dos usuários em relação a um tema específico (XIANG; ZHOU, 2014; ALAMSYAH et al., 2018)

No contexto de redes sociais, a análise de sentimentos é amplamente utilizada para monitorar reputações, prever crises e compreender o engajamento dos usuários em tempo real (JEONG; YOON; LEE, 2019). Por exemplo, estudos já demonstraram como essa técnica pode identificar tópicos polêmicos e avaliar a aceitação de serviços ou instituições (ALAMSYAH et al., 2018).

No presente trabalho, a análise de sentimentos desempenha papel essencial ao separar os tweets em três categorias (positivos, negativos e neutros), servindo como base para a aplicação da Modelagem de Tópicos. Essa combinação de técnicas possibilita não apenas identificar os principais temas discutidos, mas também compreender a carga emocional associada a eles, o que amplia a capacidade de interpretar as percepções dos estudantes em relação aos Institutos Federais.

4 Metodologia

Este estudo adota uma abordagem de pesquisa quantitativa com o objetivo de fornecer evidências empíricas robustas sobre a capacidade de gerar tópicos pertinentes e representativos com base nos dados dos Institutos Federais.

O presente trabalho aborda a estratégia de implementação da metodologia de Descoberta de Conhecimento em Bases de Dados (*Knowledge Discovery in Databases*) (CASTRO; FERRARI, 2017). Este processo segue uma série de etapas, conforme ilustrado na Figura 29, as quais serão detalhadamente exploradas ao longo desta seção.

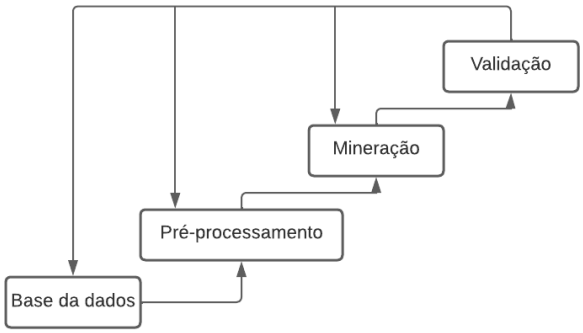


Figura 29 – Metodologia de descoberta de conhecimento em bases de dados. Fonte: (CASTRO; FERRARI, 2017)

Para a execução dos experimentos, utilizou-se a linguagem de programação *Python* em ambiente *Jupyter Notebook*, que possibilitou a organização dos códigos e análises em um formato interativo e reprodutível [1]. As simulações foram processadas em uma GPU NVIDIA Tesla P100 (PCIe, 16 GB de memória HBM2, 3584 núcleos CUDA).

4.1 Base de dados

A coleta de dados foi realizada na plataforma da rede social Twitter. Essa escolha se deve à ampla utilização do Twitter e à vasta quantidade de dados disponíveis, permitindo uma análise abrangente e detalhada da percepção e discussão em torno dos Institutos Federais.

A coleta de dados foi realizada por meio de uma raspagem contínua (scraping) de informações no Twitter, utilizando a biblioteca Selenium[2]. Esta biblioteca permite a automação da interação com páginas web, sendo especialmente eficaz para a extração de

[1] A implementação do trabalho está disponível no link<https://github.com/pesquisa-topicos-if/modelagem-topicos-if>

[2] Disponível em: https://pypi.org/project/selenium

dados de sites dinâmicos, como o Twitter, onde os conteúdos são carregados de maneira assíncrona. A implementação do Selenium possibilita a navegação automatizada e a extração eficiente dos tweets relevantes para o desenvolvimento do estudo.

A implementação foi baseada no projeto disponível em selenium-twitter-scraper[3], que fornece um modelo eficiente para a coleta de tweets. Adaptamos o código para atender às necessidades específicas da pesquisa, garantindo a obtenção de dados relevantes sobre os Institutos Federais do Brasil.

O processo de coleta inicia-se com a configuração do Selenium, onde utilizamos um WebDriver compatível com o navegador escolhido (Firefox). Esse WebDriver permite o carregamento dinâmico da página do Twitter, possibilitando a busca de tweets por meio de consultas específicas contendo as siglas dos Institutos Federais. Para capturar o máximo de tweets disponíveis, o Selenium automatiza a rolagem da página, garantindo uma coleta mais abrangente. As siglas dos IFs foram utilizados como consulta na plataforma Twitter.

Tabela 1 – Quantidade de tweets pelos Institutos Federais do Brasil

Norte			**Centro-Oeste**		
Nome	**Sigla**	**Tweets**	**Nome**	**Sigla**	**Tweets**
Acre	IFAC	676	Brasília	IFB	1001
Amapá	IFAP	1325	Mato Grosso	IFMT	3
Amazonas	IFAM	2666	Mato Grosso do Sul	IFMS	320
Pará	IFPA	1023	Goiás	IFG	1651
Rondônia	IFRO	1212	Goiano	IFGoiano	14
Tocantins	IFTO	481			
Roraima	IFRR	309			
Sudeste			**Sul**		
Rio de Janeiro	IFRJ	1864	Rio Grande do Sul	IFRS	14
Fluminense	IFF	8979	Farroupilha	IFFarroupilha	1
Minas Gerais	IFMG	670	Sul-Rio-Grandense	IFSUL	1422
Norte MG	IFNMG	93	Paraná	IFPR	827
Sudeste MG	IFSUDESTE	0	Santa Catarina	IFSC	6024
Sul MG	IFSULDEMINAS	1	Catarinense	IFC	1402
Triângulo MG	IFTM	533			
São Paulo	IFSP	2048			
Nordeste					
Alagoas	IFAL	3408	Bahia	IFBA	6979
Baiano	IF Baiano	112	Ceará	IFCE	2006
Maranhão	IFMA	3722	Paraíba	IFPB	811
Pernambuco	IFPE	896	Sertão PE	IF Sertão PE	9
Piauí	IFPI	2346	Rio Grande do Norte	IFRN	2383
Sergipe	IFS	2170			

[3] Disponível em: https://github.com/godkingjay/selenium-twitter-scraper.

A coleta de dados foi realizada no período compreendido entre agosto de 2023 e agosto de 2024. O banco de dados resultante contém aproximadamente 66.000 tweets coletados de diferentes Institutos Federais durante o processo. As siglas dos IFs distribuídos por todo o Brasil[4] e a quantidade de tweets coletados para cada sigla é apresentada na Tabela 1.

4.2 Pré-Processamento

Após a coleta dos dados, utilizando a raspagem contínua, foi construída a etapa de pré-processamento para garantir a qualidade e a consistência dos dados obtidos. Nesta fase, foram realizadas diversas operações para limpar e preparar os dados para análise. As principais etapas de pré-processamento incluiram:

- **Manter a base em Português**: A abordagem adotada para garantir que a base de dados fosse composta exclusivamente por conteúdo em português consistiu na detecção automática do idioma de cada linha do dataset. Para esse propósito, foi utilizada a biblioteca *langdetect*[5] para identificar o idioma das mensagens, filtrando exclusivamente aquelas classificadas como sendo em português.
- **Remoção de Duplicatas e Retweets**: Para garantir a representatividade dos dados, foram eliminados os retweets e mensagens duplicadas para evitar redundâncias.
- **Limpeza de Dados**: Nesta etapa, foram aplicadas técnicas de limpeza para remover ruídos e informações irrelevantes, como links, hashtags e menções, visando garantir a qualidade dos dados.
- **Tokenização e Normalização**: Os tweets foram divididos em palavras individuais por meio da tokenização e, em seguida, normalizadas para minúsculas, removendo caracteres especiais e pontuação, facilitando a análise.
- **Remoção de Stopwords**: Foram removidas palavras comuns e pouco significativas, como “e”, “ou” e “de”, dos textos, focando nas palavras mais relevantes para a análise.
- **Vetorização**: Os tweets pré-processados serão convertidos em vetores numéricos adequados para análise, permitindo a aplicação de técnicas de machine learning.
- **TF-IDF (Term Frequency-Inverse Document Frequency)**: Foi calculado o peso TF-IDF para cada palavra em cada tweet, considerando sua frequência no tweet individual e a sua importância relativa no conjunto de dados. Esse método foi

[4] Disponível em: Lista de Institutos Federais do Brasil.
[5] Disponível em: https://pypi.org/project/langdetect/

utilizado para a preparação dos dados em técnicas de modelagem de tópicos, como LDA e NMF.

- **Ajuste Fino (Fine-Tuning):** Foi realizada a etapa de ajuste fino para a classificação de sentimentos, segmentando a base de dados em três categorias: positivo, negativo e neutro. Utilizou-se um conjunto inicial de aproximadamente 2.000 registros, cujos sentimentos foram classificados manualmente, considerando o contexto do IFG e da região Centro-Oeste. Com essa base previamente rotulada, utilizamos o conjunto de toda nossa base de 66.000 registros para testes, permitindo a inferência de sentimentos com base no modelo treinado.

  Para essa tarefa, adotou-se o modelo BERTweet-BR (CARNEIRO et al., 2024), uma versão recente e especializada do modelo BERT, desenvolvida especificamente para o processamento de textos em português no contexto do Twitter. O modelo BERTweet-BR foi treinado com uma grande quantidade de dados provenientes de postagens em português no Twitter, o que lhe confere um conhecimento robusto sobre a forma como as pessoas se expressam nessa plataforma. A estratégia de ajuste fino foi essencial para a análise detalhada dos tópicos classificados em cada uma das categorias.

4.3 Mineração

Este trabalho inclui uma etapa de mineração de dados voltada para a modelagem de tópicos. Nesse processo, serão aplicadas técnicas e algoritmos para identificar e extrair tópicos. A configuração das técnicas de modelagem LDA, NMF e BERTopic, utilizadas no presente trabalho, são apre sentadas a seguir.

4.3.1 LDA (Latent Dirichlet Allocation)

O modelo LDA (Latent Dirichlet Allocation) é uma técnica de modelagem de tópicos utilizada para identificar tópicos latentes em uma coleção de textos (BLEI; NG; JORDAN, 2003). Neste trabalho, o modelo foi implementado com a biblioteca `Gensim` [6], que fornece ferramentas para análise de tópicos.

- **Corpus e Dicionário:** O corpus foi criado a partir do texto tokenizado e transformado em uma representação de bag-of-words(BoW) utilizando o `Gensim`. O dicionário, que mapeia as palavras para seus IDs correspondentes, também foi gerado com o `Gensim`.

[6] Fonte: Gensim disponível em: <https://radimrehurek.com/gensim/models/ldamodel.html>

- **Número de tópicos:** O número de tópicos foi um parâmetro ajustável, variando de 10 a 200 tópicos. A seleção do número ideal de tópicos foi baseada na avaliação da diferentes métricas dos tópicos gerados.
- **Número de palavras por tópico:** Para cada tópico, foram extraídas as 25 palavras mais representativas.
- **Validação Cruzada:** A validação cruzada foi realizada utilizando o método K-Fold, com 5 divisões. Esse processo permite avaliar o desempenho do modelo em diferentes subconjuntos dos dados, proporcionando uma análise mais robusta das configurações de tópicos testadas.

4.3.2 NMF

A extração de tópicos por meio de NMF (Non-Negative Matrix Factorization) foi realizada utilizando uma matriz TF-IDF, gerada a partir dos textos tokenizados. O modelo NMF foi implementado com a biblioteca `scikit-learn` [7], que oferece uma implementação eficiente para fatoração de matrizes não-negativas.

- **Vetor TF-IDF:** A matriz TF-IDF foi criada a partir dos textos tokenizados. Para isso, foi utilizado o `TfidfVectorizer` do `scikit-learn`, que transformou os textos em uma matriz esparsa de termos, considerando as 1000 palavras mais frequentes, com base na importância relativa de cada termo dentro do conjunto de documentos.
- **Número de tópicos:** O número de tópicos foi definido variando de 10 a 200.
- **Número de palavras por tópico:** Para cada tópico extraído, foram selecionadas as 25 palavras com maior peso, com base nos coeficientes da decomposição NMF.
- **Dicionário e Corpus:** O dicionário foi criado com o `Gensim`, e o corpus foi gerado pela transformação dos textos em uma bag-of-words (BoW), o que permitiu a extração de tópicos a partir dos dados tokenizados.
- **Estado Aleatório:** Um valor aleatório foi gerado entre 0 e 10.000 para garantir variabilidade entre as execuções do modelo, evitando a fixação de um padrão nos resultados obtidos.
- **Validação Cruzada:** A validação cruzada foi realizada utilizando o método K-Fold, com 5 divisões. Esse processo permite avaliar o desempenho do modelo em diferentes subconjuntos dos dados, proporcionando uma análise mais robusta das configurações de tópicos testadas.

[7] Fonte: Biblioteca `scikit-learn` disponível em <https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.NMF.html>

4.3.3 BERTopic

A modelagem de tópicos com BERTopic foi realizada conforme a metodologia proposta pelos autores (GROOTENDORST, 2022), utilizando os algoritmos HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise) para agrupamento e UMAP (Uniform Manifold Approximation and Projection) para redução de dimensionalidade. A biblioteca BERTopic [8] utiliza um modelo de embeddings de sentenças para capturar as relações semânticas entre os textos, possibilitando uma análise de tópicos mais precisa e informativa

Os parâmetros utilizados na modelagem de tópicos foram os seguintes:

- **Modelos de Embedding:**
  - `all-mpnet-base-v2:` Baseado no modelo MPNet, que combina técnicas de aprendizado para melhorar a compreensão do significado das frases. Ele gera embeddings mais precisos, capturando melhor as relações semânticas entre sentenças, sendo útil para tarefas como comparação de textos, busca por similaridade e classificação de textos [9].
  - `neuralmind/bert-large-portuguese-cased`: Versão do BERT treinada especificamente para o português, conhecida como BERTimbau (SOUZA; NOGUEIRA; LOTUFO, 2020) [10]. Utilizamos este modelo para gerar embeddings mais ajustados à língua portuguesa.
  - `melll-uff/bertweetbr:` Modelo baseado no BERTweet (CARNEIRO et al., 2024), treinado para o português brasileiro, otimizado para tarefas de PLN relacionadas a redes sociais e textos informais[11].
  - `paraphrase-multilingual-mpnet-base-v2:` Modelo de embeddings treinado para identificação de paráfrases em múltiplos idiomas, proporcionando melhor generalização em contextos multilíngues [12].
  - `FineTunning-BERTweet-BR:` Modelo otimizado por meio de fine-tuning em um conjunto de dados específico para análise de sentimentos.
- **Número de Tópicos:** Utilizado um range de valores de 10 a 200 tópicos.

[8] Fonte: Biblioteca BERTopic disponível em <https://github.com/MaartenGr/BERTopic>, (GROOTENDORST, 2022)
[9] Fonte: HuggingFace, disponível em: <https://huggingface.co/sentence-transformers/all-mpnet-base-v2>
[10] Fonte: HuggingFace, disponível em: <https://huggingface.co/neuralmind/bert-large-portuguese-cased>
[11] Fonte: HuggingFace, disponível em: <https://huggingface.co/melll-uff/bertweetbr>
[12] Fonte: HuggingFace, disponível em: <https://huggingface.co/sentence-transformers/paraphrase-multilingual-mpnet-base-v2>

- **Número de palavras por tópico:** 25 palavras principais extraídas de cada conjunto de tópicos.
- **Validação Cruzada:** K-Fold com 5 divisões para avaliar diferentes configurações de tópicos.

4.4 Validação

Nesta seção, apresentaremos as técnicas de validação em *machine learning* que foram utilizadas em nosso trabalho. Analisaremos métodos fundamentais que garantirão a robustez e confiabilidade dos resultados obtidos em nossas análises.

- **Coerência de Tópicos**: Esta métrica foi utilizada para avaliar a interpretabilidade dos tópicos gerados, medindo a similaridade semântica entre as palavras-chave de um tópico. Espera-se que uma coerência maior indique tópicos mais interpretáveis e relevantes. A equação 4.1 expressa o cálculo da métrica de coerência de tópicos C_V:

  A coerência $C_v(t)$ de um tópico t é dada por:

$$C_v(t) = \frac{1}{N} \sum_{i=1}^{N} \sum_{j=i+1}^{N} \text{NPMI}(w_i, w_j) \tag{4.1}$$

  onde:

  - C_V: representa a coerência do tópico, baseada na similaridade semântica entre as palavras do tópico.
  - w_i, w_j: são palavras dentro do conjunto de palavras mais prováveis para o tópico.
  - $\text{NPMI}(w_i, w_j)$: medida de similaridade normalizada entre as palavras w_i e w_j, calculada utilizando a informação mútua pontual normalizada (NPMI).
  - N: número total de palavras no tópico.

  A métrica C_V é baseada em uma janela deslizante, uma segmentação de um único conjunto das palavras principais e uma medida de confirmação indireta (RöDER; BOTH; HINNEBURG, 2015). Utilizando a NPMI e a similaridade cosseno, essa métrica calcula a coerência do tópico ao avaliar a coocorrência das palavras dentro de documentos e a similaridade semântica entre elas. Quanto maior o valor de C_V, maior a coerência do tópico, o que indica que o tópico é mais interpretável e relevante. A abordagem leva em consideração as coocorrências das palavras dentro do tópico e o cálculo da similaridade entre os vetores representando as palavras, utilizando a similaridade cosseno.

- **Visualização das Métricas de Modelagem**: Gráficos com intervalo de confiança foram empregados para visualizar e avaliar várias métricas de desempenho dos tópicos, incluindo coerência, diversidade, estabilidade e tempo computacional. Utilizando a biblioteca `matplotlib`, plotamos gráficos que apresentam as médias dessas métricas, com barras de erro representando os intervalos de confiança. Isso permitiu uma análise mais robusta das variações e incertezas associadas aos resultados para diferentes números de tópicos e métodos de modelagem. As métricas são avaliadas em função do número de tópicos, fornecendo uma visão detalhada da eficácia e confiabilidade de cada modelo ao longo da quantidade de tópicos.

- **Validação Cruzada (Cross-Validation)**: A validação cruzada k-fold (REFAEILZADEH; TANG; LIU, 2009), conforme ilustrada na Figura 30 foi utilizada para avaliar os modelos de aprendizado. De forma análoga, em nosso trabalho, o conjunto de dados foi dividido em 5 partes, e o modelo foi treinado 5 vezes, utilizando uma dobra (*fold*) diferente para teste em cada iteração. Essa abordagem garantiu uma avaliação mais robusta e representativa do desempenho, com cada ponto de dado sendo usado tanto para treinamento quanto para validação. A média das métricas de desempenho, como coerência, diversidade, estabilidade e tempo computacional, foi calculada para fornecer uma estimativa precisa e estável do modelo.

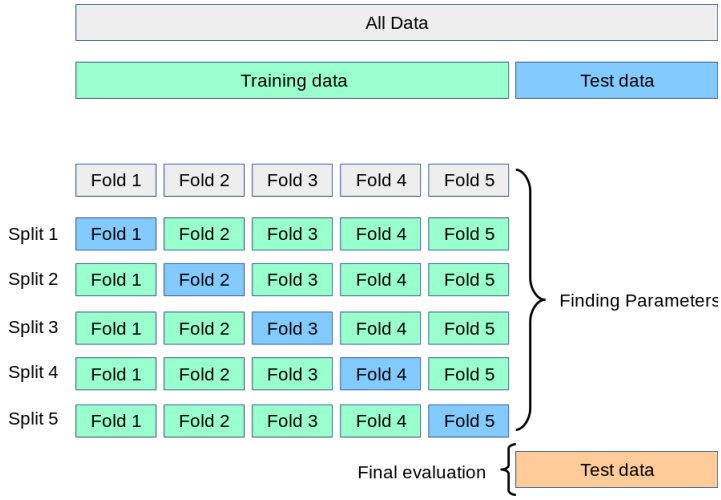


Figura 30 – K-fold Cross-validation. Fonte: (REFAEILZADEH; TANG; LIU, 2009)

- **Diversidade** A diversidade de tópicos é uma métrica que avalia o quão distintos semanticamente são os tópicos gerados por um modelo de tópicos. Segundo Dieng, Ruiz e Blei (2020), uma forma de medir essa diversidade é calcular a porcentagem de palavras únicas entre as 25 principais palavras de todos os tópicos.

  Neste trabalho, seguimos essa abordagem e consideramos as 25 palavras mais relevantes de cada tópico para calcular a diversidade. Um modelo eficaz deve gerar tópicos diversificados, resultando em uma pontuação alta nessa métrica. Caso contrário, uma

baixa diversidade indica redundância entre os tópicos, sugerindo que o modelo não conseguiu distinguir suficientemente os temas presentes no corpus.

- **Estabilidade – Rank Biased Overlap (RBO)**

  Para avaliar a estabilidade dos tópicos gerados em diferentes execuções do modelo, utilizamos a métrica *Rank Biased Overlap* (RBO), conforme proposta por Webber, Moffat e Zobel (2010). O RBO mede a similaridade entre listas ordenadas, permitindo avaliar a consistência das palavras-chave que definem cada tópico ao longo das execuções.

  A métrica é definida pela equação:

$$RBO(T_1, T_2, p, d) = \frac{x_d}{d} \cdot p^d + \frac{1-p}{d} \sum_{i=1}^{d} \frac{x_i}{i} \cdot p^i \tag{4.2}$$

  Onde:

  - T_1 e T_2 são listas ordenadas com as n palavras principais de dois tópicos em diferentes execuções;
  - d é a profundidade da comparação, normalmente definida como 10 palavras principais;
  - x_d representa a interseção entre T_1 e T_2 até a profundidade d;
  - p controla o impacto da ordem das palavras na métrica.

  O parâmetro p controla a influência da posição das palavras na métrica. Quando $p = 1$, a métrica considera exclusivamente a interseção das palavras, desconsiderando sua ordem. Por outro lado, valores menores de p atribuem maior peso à posição das palavras na lista. Conforme recomendado por Webber, Moffat e Zobel (2010), foi adotado $p = 0.9$, um valor que proporciona um equilíbrio adequado entre a interseção e a ordem das palavras. Além disso, foi definido $d = 10$, uma vez que a comparação é realizada com as 10 palavras mais representativas de cada tópico, além disso, definimos a quantidade de 5 execuções para a estabilidade. Utilizamos a biblioteca *RBO*[13] para implementar essa abordagem.

- **Tempo Computacional**

  O treinamento dos modelos foi executado em uma GPU NVIDIA Tesla P100 (PCIe, 16 GB de memória HBM2, 3584 núcleos CUDA). O tempo computacional foi definido como o intervalo entre o início e o término do treinamento dos modelos de aprendizado de máquina para modelagem de tópicos, sendo medido por meio do registro do tempo inicial e final do processo, calculando a diferença entre ambos.

[13] RBO disponível em: <https://github.com/changyaochen/rbo>

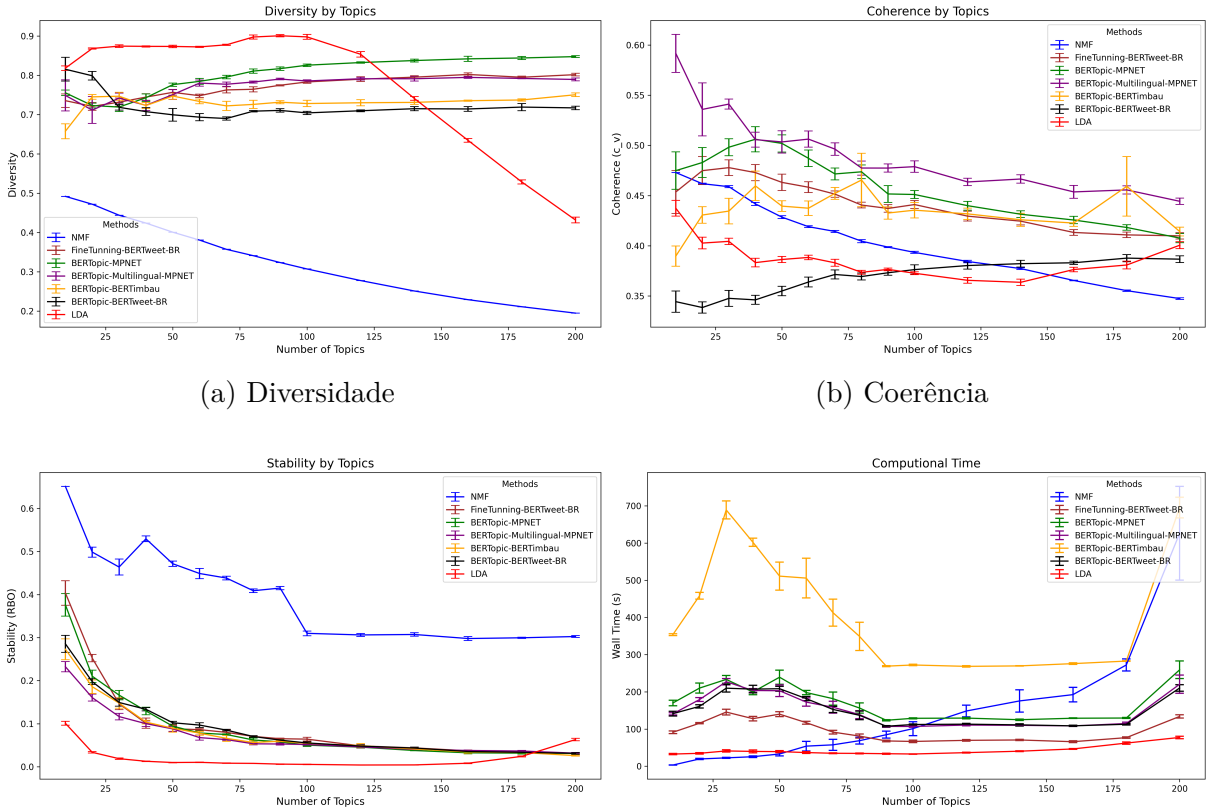
Para garantir maior precisão e evitar efeitos de variação em execuções individuais, o treinamento foi realizado em diferentes configurações de número de tópicos e validado por meio de um procedimento de validação cruzada com k=5 *folds*. Os tempos de treinamento foram registrados para diferentes quantidades de tópicos, variando de 10 a 200 tópicos.

5 Resultados Experimentais

Nesta seção, serão apresentados os resultados dos modelos considerando diferentes perspectivas. Inicialmente, serão discutidos os resultados da análise dos tópicos gerais. Em seguida, a análise dos resultados será segmentada por sentimento. Por fim, será realizada uma avaliação dos tópicos com base em sua distribuição regional.

5.1 Modelagem de Tópicos com Todos os Tweets

Para a comparação dos modelos, foram adotadas quatro métricas principais: diversidade, coerência, estabilidade e tempo computacional. Essas abordagens permitiram uma avaliação abrangente do desempenho de cada modelo. No total, analisamos sete modelos, incluindo LDA, NMF e abordagens baseadas em Transformers, como o BERTopic. Além disso, aprimoramos um modelo otimizado por meio de ajuste fino, denominado FineTuning-BERTweet-BR.



(a) Diversidade

(b) Coerência

(c) Estabilidade

(d) Tempo computacional

Figura 31 – Comparação das métricas.

De forma geral, não foi identificado um único método que se destacasse em todas

as métricas de Modelagem de Tópicos, uma vez que cada métrica apresentou um modelo com melhor desempenho. Os resultados do modelo FineTuning-BERTweet-BR indicaram melhorias significativas, especialmente em relação à diversidade e coerência dos tópicos gerais, quando comparado à sua versão não ajustada, BERTopic-BERTweet-BR. O modelo BERTopic-Multilingual-MPNET obteve melhor desempenho na métrica de coerência, enquanto o modelo BERTopic-MPNET se destacou na diversidade. Além disso, o modelo NMF demonstrou maior estabilidade, enquanto o tempo computacional dos modelos analisados apresentou apenas pequenas variações. Nas subseções a seguir, cada métrica será detalhada, juntamente com a apresentação dos respectivos resultados.

5.1.1 Diversidade

A diversidade dos tópicos varia conforme o modelo utilizado, como ilustrado na Figura 31a. O LDA inicialmente apresenta alta diversidade, mas essa diversidade diminui à medida que o número de tópicos aumenta, devido a não utilizar os embeddings contextuais, e portanto não é possível incluir palavras raras em Tópicos que possuem relação semântica com tais palavras. O NMF, por sua vez, apresenta baixa diversidade, pois não possui uma etapa de clusterização, mas descobre os tópicos a partir da fatoração de matrizes em uma única etapa, o que resulta em maior sobreposição entre tópicos.

Os modelos baseados em BERTopic mantêm alta diversidade mesmo com um grande número de tópicos, pois são capazes de identificar relações semânticas entre termos raros, agrupando-os de forma mais coerente (GROOTENDORST, 2022). Além disso, o modelo FineTuning-BERTweet-BR evidencia uma melhoria significativa na captura da diversidade em comparação à sua versão sem ajustes, BERTopic-BERTweet-BR.

5.1.2 Coerência

Os modelos baseados em BERT conforme apresentado na Figura 31b também apresentam maior coerência em relação ao LDA e NMF, pois utilizam embeddings para capturar o contexto e o significado das palavras, enquanto os métodos tradicionais dependem apenas da contagem de palavras, ignorando suas relações semânticas. Com técnicas avançadas de redução de dimensionalidade, como UMAP, e agrupamento eficiente via HDBSCAN, o BERTopic gera tópicos mais bem estruturados e semanticamente coerentes. Como resultado, os tópicos extraídos apresentam maior relevância e melhor organização, elevando a métrica de coerência.

O BERTopic-BERTweet apresentou os piores resultado de coerência em relação ao demais métodos baseados em BERT, pois ele foi treinado em coleções de tweets com pouca ênfase em textos objetivos encontrados em tweets neutros que normalmente estão associados a informes do setor de comunicação dos IFs. Entretanto, a tunagem fina do BERTweet

(FineTunning-BERTweet-BR) apresentou resultados significativamente superiores a versão sem tunagem fina (BERTopic-BERTweet-BR), reforçando a necessidade de ajuste para o aprimoramento do desempenho do modelo.

5.1.3 Estabilidade

O NMF conforme mostrado na Figura 31c, apresenta maior estabilidade RBO em comparação ao LDA e BERTopic porque é um modelo determinístico, garantindo que os tópicos extraídos permaneçam consistentes entre diferentes execuções. Em contraste, o LDA, por ser probabilístico e depender de inicializações aleatórias, gera variações nos tópicos extraídos, reduzindo sua estabilidade. Já o BERTopic, que utiliza embeddings contextuais e técnicas como UMAP e HDBSCAN para agrupamento, é altamente sensível a pequenas variações nos dados, tornando seus tópicos menos estáveis em execuções repetidas em diferentes amostras dos dados. Assim, a ausência de aleatoriedade no NMF permite que ele alcance maior estabilidade em comparação aos outros modelos.

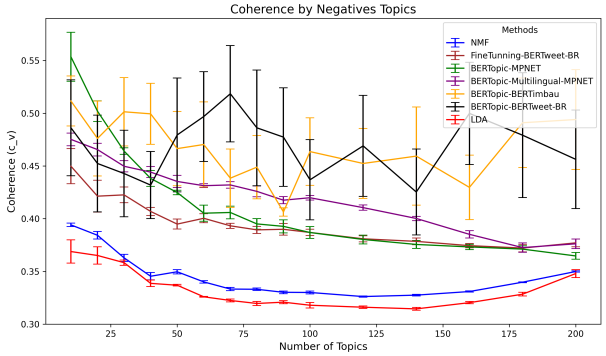
5.1.4 Tempo computacional

O gráfico 31d mostra que o tempo computacional dos modelos baseados em BERT é mais elevado no início, quando há poucos tópicos, e diminui conforme o número de tópicos aumenta. Isso ocorre porque, no início, o modelo precisa realizar mais cálculos para definir representações iniciais e estabelecer relações semânticas entre termos, o que exige maior esforço computacional. À medida que mais tópicos são adicionados, os embeddings já estão melhor estruturados, e os algoritmos de redução de dimensionalidade (UMAP) e agrupamento (HDBSCAN) conseguem processar os dados de forma mais eficiente, reduzindo o tempo de execução.

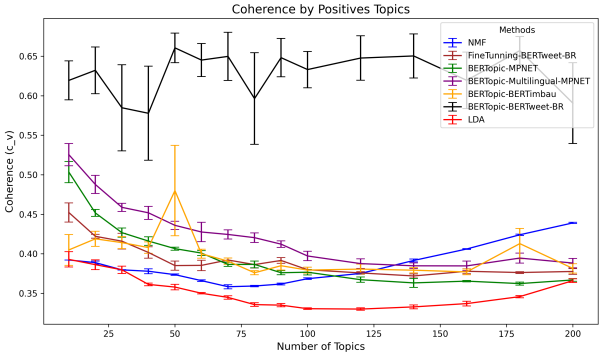
Já o NMF apresenta um aumento no tempo de execução conforme o número de tópicos cresce devido à sua abordagem matemática para decomposição da matriz de termos e documentos. No entanto, em comparação com outros modelos como BERTopic-MPNet, BERTopic-Multilingual-MPNET, BERTopic-BERTweet e FineTunning-BERTweet-BR, o BETopic-BERTimbau ainda mantém um tempo computacional mais alto, possivelmente devido a dificuldade de convergência dos clusters que representam os tópicos. Tal dificuldade se deve a diversidade de representações de termos obtida durante o pré-treinamento no conjunto de páginas da web da coleção brWaC (FILHO et al., 2018), e desta forma, tornam a extração de tópicos mais custosa computacionalmente, explicando o desempenho inferior em termos de tempo de processamento.

5.2 Modelagem de Tópicos em Tweets Separados por Sentimento

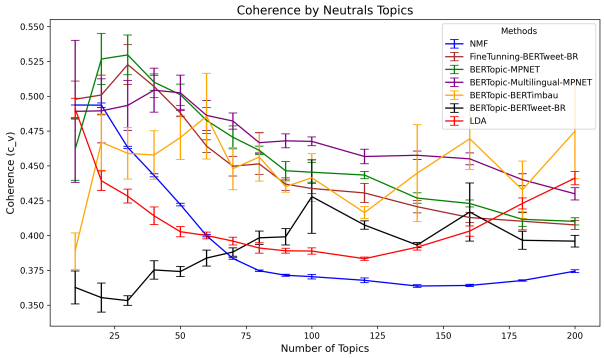
Para a identificação dos tópicos, a coleção foi inicialmente dividida em três categorias de sentimento, conforme descrito na seção 4.2. A seguir, são apresentados os métodos juntamente com seus respectivos valores de coerência nas diferentes separações da coleção, considerando tweets positivos, negativos e neutros.



(a) Coerência: Tópicos negativos



(b) Coerência: Tópicos positivos



(c) Coerência: Tópicos neutros

Figura 32 – Comparação das métricas de coerência por sentimento.

Conforme ilustrado na Figura 32, os modelos BERTopic-BERTweet-BR e BERTopic-BERTimbau demonstraram desempenho superior na classificação de sentimentos positivos e negativos, uma vez que foram pré-treinados especificamente para essa tarefa. Por outro lado, o modelo BERTopic-Multilingual-MPNET obteve melhor eficiência na classificação de sentimentos neutros, pois é mais eficaz na compreensão de contextos gerais usualmente presente nestes tipos de tweets. Além disso, observa-se que a versão do BERTweet com ajuste fino (FineTunning-BERTweet-BR) na coerência de Tópicos Neutros, apresenta um desempenho superior em comparação à versão sem esse ajuste, evidenciando a importância desse refinamento para a melhoria da eficiência do modelo. Com base nesses resultados, selecionamos os modelos mais adequados para a construção dos tópicos.

Para a geração dos tópicos, adotamos o modelo mais eficiente para cada categoria de sentimento, selecionando-o com base no número de tópicos que apresentou a melhor

coerência. Para os tópicos positivos, identificamos que o ponto de maior coerência ocorreu com 70 tópicos, optando, assim, pelo modelo BERTopic-BERTweet-BR. De forma semelhante, para os tópicos negativos, também empregamos o BERTopic-BERTweet-BR, definindo 50 tópicos, valor que resultou na melhor coerência observada. Já para os tópicos neutros, adotamos o modelo BERTopic-MPNET, determinando 40 tópicos, número que apresentou a maior coerência. Para facilitar a visualização e a interpretação dos resultados, apresentamos os 20 tópicos mais relevantes (dentre os tópicos gerados) e as três palavras mais relevantes por tópico, conforme a abordagem adotada por (HELLWIG et al., 2024).

Tabela 2 – Distribuição dos tópicos por sentimento

Tópicos Negativos	**Tópicos Positivos**	**Tópicos Neutros**
greve, aluno, professores	campus, educação, estudantes	prova, amanhã, gente
matar, passar, culpa	saudades, pessoas, época	inscrições, vagas, cursos
férias, faltam, piscina	passei, simplesmente, povo	aula, professor, ensino
questões, prova, matemática	professores, alunos, queridos	menino, mãe, pai
greve, internet, acabou	campus, melhor, maior	campus, construção, lula
sair, desistir, pior	passei, lugar, primeiro	almoço, café, comer
prova, estudando, nervosa	estudante, sonho, maior	foto, twitter, instagran
aulas, semanas, voltam	saudades, real, dizer	biblioteca, livros, eleição
mãe, casa, ficar	passei, entrar, preciso	música, global, albúm
médio, ensino, semestre	engenharia, química, elétrica	sonhei, noite, dormir
povo, doido, fofoqueiro	prova, acertei, questões	ônibus, acidente, motorista
deus, estudante, prova	sentir, tanta, saudades	química, biologia, física
internet, site, matar	deus, abençoe, graças	banheiro, praia, piscina
vida, pior, triste	meninas, amigas, bonitas	camisa, cabelo, uniforme
resultado, inscrição, gabarito	coisas, pessoas, sempre	greve, sinasefe, mobilização
mundo, odeia, bloquear	amor, obrigado, amiga	festa, carnaval, junina
sair, indecisão, vontade	internet, resenha, boa	computadores, software, site
fraco, aluno, professor	campo, melhores, disparado	igreja, oração, rezar
site, funciona, resultado	amigo, saudades, bateu	engenharia, edificações, arquitetura
seriamente, pensando, desistir	prova, aventurar, sonhador	dinheiro, ingresso, comprar

A análise da distribuição dos tópicos por sentimento, apresentada na Tabela 2, evidencia padrões distintos nas discussões analisadas. Nos tópicos classificados como negativos, observa-se uma diversidade de insatisfações relacionadas ao contexto dos Institutos Federais (IFs). Termos como *greve*, *aluno* e *professores* aparecem fortemente associados às discussões sobre paralisações e seus impactos. Já expressões como *prova*, *estudando* e *nervosa* refletem a pressão e o nervosismo relacionados à rotina acadêmica, especialmente em momentos de avaliação. Esse aspecto pode ser ilustrado por um tweet como: *"queria fingir que nao to extremamente nervosa pra prova do iff"*, que exemplifica o vínculo entre o desempenho estudantil e o estresse emocional identificado nos tópicos.

Além disso, observam-se menções relacionadas à evasão estudantil, refletidas em tópicos compostos por termos como *sair*, *indecisão* e *vontade*, bem como em tópicos

formados por *seriamente*, *pensando* e *desistir*.

Esses indícios sugerem que parte dos estudantes expressa abertamente a intenção de abandonar os cursos. Um exemplo desse padrão pode ser visto em um tweet como: *"to pensando seriamente em desistir do ifsc, meu psicológico tá f..."*[1], no qual se evidencia o impacto emocional e psicológico associado à permanência acadêmica.

Esses padrões podem ser visualizados de maneira mais clara na nuvem de palavras da Figura 33, onde se destacam os termos mais frequentes nos tweets classificados como negativos. A visualização evidencia os desafios acadêmicos quanto as percepções sociais que permeiam as experiências relatadas pelos estudantes.

Figura 33 – Nuvem de palavras representando os Tópicos Negativos.

Por outro lado, entre os tópicos classificados como positivos, destacam-se aqueles associados a termos como *passei*, *lugar* e *primeiro*, frequentemente relacionados à conquista de uma vaga em processos seletivos ou no ingresso em universidades vinculadas aos Institutos Federais. Esse padrão pode ser observado em mensagens como: *"Passei na seleção do IFCE em primeiro lugar"*, que refletem sentimentos de orgulho e realização pessoal. Além disso, emergem tópicos relacionados à rotina de estudos e avaliações, evidenciados por termos como *prova*, *acertei* e *questões*. Um exemplo é o tweet: *"Tentei a prova do IFCE (português) e só acertei uma questão; fiz de novo agorinha (depois de um tempo, claro) e acertei três. Estou evoluindo, tá!"*, que ilustra a percepção de progresso acadêmico e motivação.

Outro conjunto de tópicos positivos refere-se a sentimentos de nostalgia, com a recorrência de termos como *saudades*, *pessoas* e *época*, bem como de tópicos formados por palavras como *sentir*, *tanta* e *saudades*, revelando a memória afetiva dos estudantes em relação à sua vivência nos IFs.

Esses padrões podem ser visualizados na Figura 34, que apresenta a nuvem de palavras referentes aos tópicos classificados como positivos.

[1] O termo original utilizado no tweet contém linguagem ofensiva, omitida neste trabalho para preservar a formalidade do texto científico.

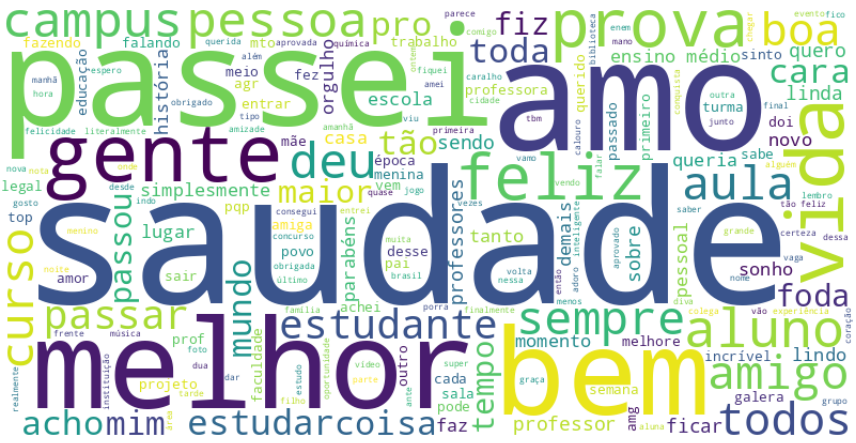


Figura 34 – Nuvem de palavras representando os Tópicos Positivos.

Em relação aos tópicos classificados como neutros, observa-se que estes abrangem, predominantemente, conteúdos relacionados à comunicação institucional, divulgação de informações e eventos. No âmbito da comunicação institucional, destaca-se, por exemplo, o tópico formado por termos como *inscrições*, *vagas* e *cursos*, frequentemente associado a anúncios oficiais de oportunidades educacionais. Esse padrão pode ser ilustrado por mensagens como: *"Inscrições para cursos gratuitos do IFBA encerram no dia de julho. Há vagas para cursos de desenhista de produtos gráficos web e montador e reparador de computadores."* Outro conjunto de tópicos refere-se à divulgação e promoção de festas e eventos, nos quais se destacam termos como *festa*, *carnaval* e *junina*. Um exemplo disso pode ser observado no tweet: *"Vou dormir todos os dias sonhando com a festa junina do IFPR."*

A distribuição dos termos associados aos tópicos neutros pode ser visualizada na Figura 35, que apresenta a respectiva nuvem de palavras.

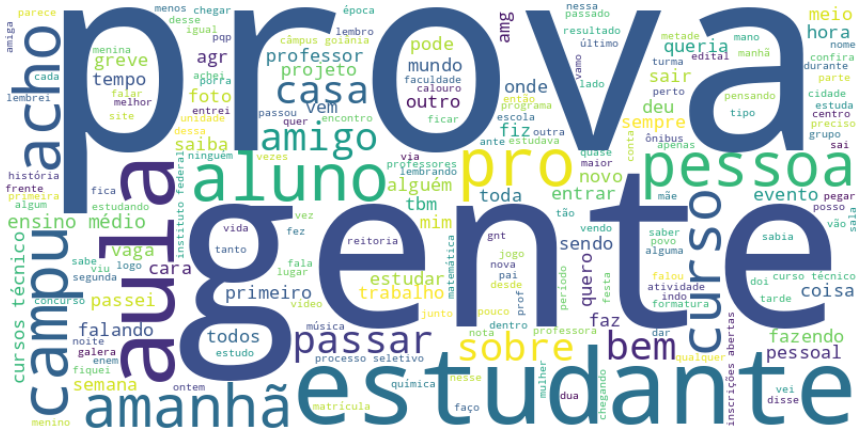


Figura 35 – Nuvem de palavras representando os Tópicos Neutros.

Além disso, também se identificaram tópicos relacionados à circulação de informes sobre a greve, com destaque para os termos *greve*, *sinasefe* e *mobilização*. Esse padrão pode ser evidenciado em publicações como: *"O Comando de Greve do IFRN Campus Natal-*

Centro Histórico, em parceria com o SINASEFE Natal, realiza na próxima segunda-feira uma programação de mobilização no campus."

5.3 Modelagem de Tópicos em Tweets Separados por Região

Para a análise dos dados regionais, é fundamental compreender a distribuição dos tweets em cada região, considerando os diferentes sentimentos atribuídos. A Tabela 3 apresenta essa distribuição, servindo como base para a análise de cada região nas subseções seguintes.

Região	**Negativos**	**Neutros**	**Positivos**
Norte	2886	3133	1249
Nordeste	9083	10208	4281
Centro-Oeste	580	1620	403
Sudeste	9490	9673	4396
Sul	3582	3867	1671
Total	25621	28501	12000

Tabela 3 – Distribuição dos tweets por região e sentimento.

De forma semelhante à geração de tópicos para os sentimentos, adotamos a estratégia de Modelagem de Tópicos com base na métrica de coerência para cada região. O modelo foi selecionado com base na maior coerência observada em relação ao número de tópicos, garantindo a escolha mais adequada para a análise. Nas próximas subseções, serão analisados os resultados obtidos. Para aprimorar a visualização e a interpretação dos dados, apresentamos os 20 tópicos mais relevantes, acompanhados das três palavras mais representativas de cada tópico, em conformidade com a visualização apresentada por Hellwig et al. (2024)

5.3.1 Modelagem de Tópicos na Região Centro-Oeste

Conforme ilustrado no gráfico da Figura 36, os modelos BERTopic-Multilingual-MPNET, BERTopic-BERTimbau, BERTopic-BERTweet-BR e o FineTunning-BERTweet-BR apresentaram desempenhos semelhantes (i.e. com intervalos de confianças sobrepostos), na maior parte do número de tópicos avaliados. Esse resultado se justifica pelo fato de que a maioria dos tópicos identificados na região Centro-Oeste são neutros, conforme indicado na Tabela 3, e, portanto, os principais temas abordados estão relacionados à comunicação institucional.

A extração dos tópicos foi realizada utilizando o modelo BERTopic-BERTimbau, uma vez que ele apresentou o maior ponto de coerência, sendo utilizado com 50 tópicos. Os tópicos identificados da Região Centro-Oeste, apresentados na Tabela 4, abrangem

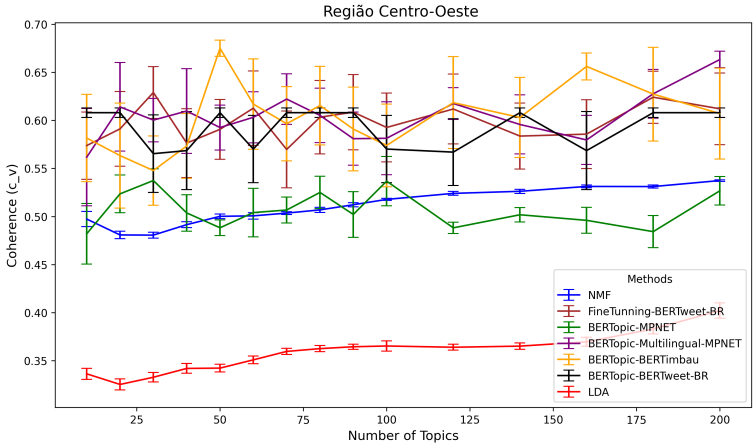


Figura 36 – Coerência: Região Centro-Oeste.

eventos institucionais, ilustrados com tópico associados a termos como *ciências, evento, tecnologia* e tópico relacionados aos termos *computação, programação, semana*. Foram também identificado tópicos relacionados a apresentações culturais, com tópico relacionadas a termos como *teatro, espetáculo, concerto*. O processos de inscrição foi representados pelo tópico com os termos *inscrições, cursos, seleção*. Em relação aos informes sobre o retorno às aulas, com o tópico relacionados aos termos *reunião, retorno, aulas*.

Tabela 4 – Distribuição dos tópicos na região Centro-Oeste

Tópicos		
greve, aula, faculdade	almoço, lanchonete, café	negra, consciência, estudos
ônibus, linha, taguatinga	reunião, retorno, aulas	inscrições, cursos, seleção
ciências, evento, tecnologia	alimentos, agricultores, pnae	técnica, visita, vídeo
mobilidade, engenharia, ciclístico	nascente, sol, fundamental	movimento, educação, infantil
dança, prática, experiência	campus, brasília, reitora	computação, programação, semana
extensão, projeto, coordenada	teatro, espetáculo, concerto	canal, youtube, prevenção
mulher, feira, economia	brasil, voto, conquistar	

5.3.2 Modelagem de Tópicos na Região Sudeste

Na região Sudeste, conforme apresentado na Tabela 3, observou-se uma distribuição significativa de tweets classificados como positivos e negativos, superando os classificados como neutros. Nesse contexto, o modelo BERTopic-BERTweet-BR demonstrou a melhor coerência em todos os números de tópicos avaliados, conforme ilustrado na Figura 37, evidenciando sua capacidade de capturar padrões distintos de polaridade de sentimentos presentes na base de dados da região. Em razão disso, este modelo foi utilizado para a geração dos tópicos, com a definição de 50 tópicos, número que apresentou a maior coerência.

Os tópicos discutidos na Região Sudeste, apresentados na Tabela 5, abrangem aspectos relacionados à rotina dos estudantes, incluindo tópico associados aos termos *ônibus, campus, prova*, outro tópico contendo os termos *estudei, prova, amanhã* e um terceiro tópico que refere-se aos termos *sorte, todos, prova*. Além disso, há menções a

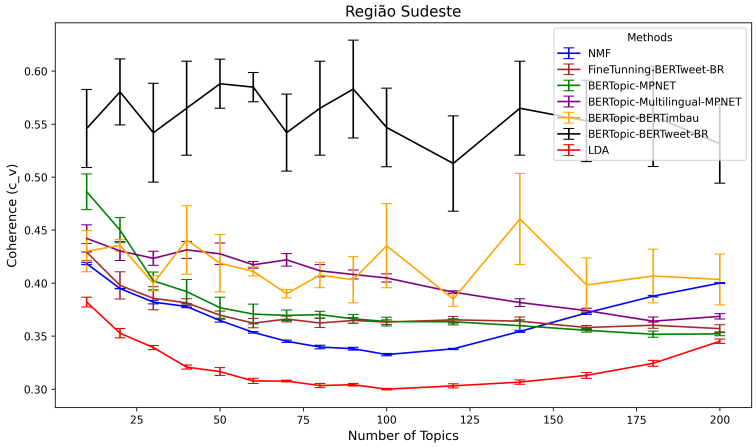


Figura 37 – Coerência: Região Sudeste.

conquistas acadêmicas, com o tópico vinculados aos termos *diploma, aprovação, festa*. A comunicação institucional sobre oportunidades acadêmicas também se faz presente, com o tópico relacionado a *vagas, especialização, gratuita*. Por fim, são observados informes sobre eventos culturais, como festas juninas, representadas pelos tópico com os termos *junina, festa, quadrilha*.

Tabela 5 – Distribuição dos tópicos na região Sudeste

Tópicos		
onibus, campus, prova	diploma, aprovação, festa	vagas, especialização, gratuita
curso, ensino, educação	passei, feliz, mundo	médio, ensino, alojamento
resultado, semana, prova	deus, mente, amor	sorte, todos, prova
professores, alunos, melhores	emprestar, pago, avisa	estudei, prova, amanhã
aulas, pessoas, voltam	saudades, sentir, falta	feliz, triste, choro
aula, engenharia, estudante	sair, estudar, dormir	junina, festa, quadrilha
passar, estudar, entrei	prova, matemática, sonhei	

5.3.3 Modelagem de tópicos na Região Nordeste

Na região Nordeste, conforme ilustrado na Figura 38, os modelos apresentaram um desempenho equilibrado. O BERTopic-BERTweet-BR e BERTopic-BERTimbau apresentaram os melhores resultados ao longo dos números de tópicos, além de obter bons resultados em conjunto com os modelos BERTopic-Multilingual-MPNET e FineTuning-BERTweet-BR. Esse desempenho pode ser atribuído à diversidade temática da base de dados conforme apresentado na Tabela 3, caracterizada por uma expressiva quantidade de tópicos neutros, o que beneficiou modelos mais generalistas na identificação desses tópicos. No entanto, a análise dos tópicos foi realizada com o modelo BERTopic-BERTweet-BR, uma vez que este apresentou a melhor coerência com 120 tópicos.

Os tópicos gerados na região Nordeste, conforme apresentado na Tabela 6, revela discussões relacionadas à rotina dos estudantes. Destacam-se o tópico associados aos termos *provas, questões, acertei* e tópicos com os termos *amanhã, domingo, prova*, além da

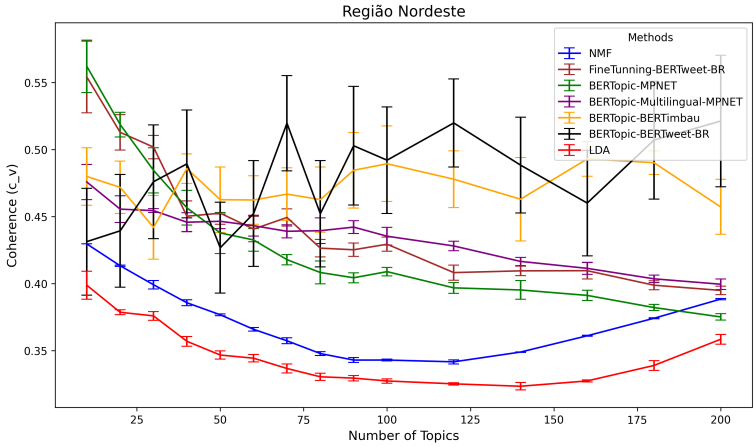


Figura 38 – Coerência: Região Nordeste.

manifestação de apreensão dos alunos em relação às avaliações, representada pelo tópico com os termos *prova, medo, aula.*

Tabela 6 – Distribuição dos tópicos na região Nordeste

Tópicos		
campus, greve, ensino	provas, questões, acertei	fraco, medo, abandonado
pessoas, aulas, meninas	greve, internet, aprovada	época, melhor, fase
povo, fraco, estudante	estudantes, debates, efeitos	mundo, correndo, estudou
passar, normal, entrei	favor, emprestar, alguém	morar, estudo, acontecer
médio, ensino, física	prova, medo, aula	passar, desmaio, acreditar
deus, senhor, mente	resultado, lista, aprovados	professor, vaga, estressado
saudades, sentir, falta	amanhã, domingo, prova	

Ademais, o tópico associado aos termos *campus*, *greve* e *ensino* reflete o debate sobre a greve de 2024 nos Institutos Federais, no contexto educacional da região. Por fim, o tópico com os termos *fraco, abandono, medo* e o tópico contendo os termos *povo, fraco, estudante* evidenciam os desafios enfrentados pelos alunos.

5.3.4 Modelagem de Tópicos na Região Norte

Na região Norte, conforme apresentado na Figura 39, o modelo BERTopic-BERTweet-BR demonstrou a melhor coerência em comparação com os demais modelos, na maioria dos números de tópicos avaliados, beneficiando-se da predominância de tweets classificados como positivos e negativos, conforme é apresentado na Tabela 3. No entanto, tanto o modelo BERTopic-Multilingual-MPNET e o FineTunning-BERTweet-BR também obtiveram resultados satisfatórios, demonstrando uma boa adaptação às características da base de dados regional.

A análise dos tópicos da Região Norte foi realizada com o modelo BERTopic-BERTweet-BR, uma vez que este apresentou o melhor resultado de coerência, com uma quantidade de 100 tópicos. Conforme demonstrado na Tabela 7, os tópicos gerados destaca

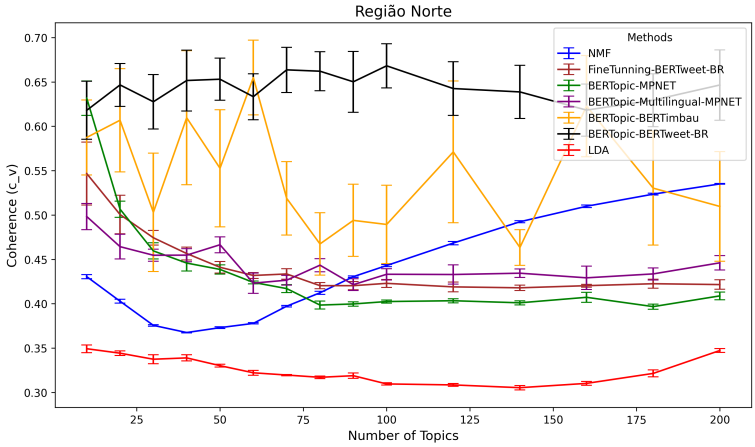


Figura 39 – Coerência: Região Norte.

questões educacionais, com ênfase na greve de 2024. Esse tema é representado pelo tópico associado aos termos *acadêmico*, *calendário* e *suspenso*, os quais indicam a paralisação das atividades acadêmicas. Além disso, observam-se tópicos relacionados à comunicação institucional sobre processos seletivos e vestibulares, como o tópico composto pelos termos *edital*, *seletivo* e *processo*, bem como outro tópico associado aos termos *vestibular*, *provas* e *vagas*.

Por fim, destaca-se o tópico composto pelos termos *acidente*, *Tucuruí* e *ônibus*, que faz referência a um trágico acidente ocorrido nas proximidades da Usina de Tucuruí, envolvendo um ônibus do IFPA. O incidente resultou em quatro mortes e diversos feridos, gerando ampla repercussão nas redes sociais [2].

Tabela 7 – Distribuição dos tópicos na região Norte

Tópicos		
passar, vida, casa	história, palestra, contar	mental, saúde, nervosismo
professor, campus, escola	palestrar, jif, cantar	vestibular, provas, vagas
educação, presidente, amazonas	halloween, festa, junina	biologia, ciências, candidaturas
foto, filme, vídeos	edital, seletivo, processo	café, restaurante, manhã
voltar, aulas, casa	concurso, técnico, curso	futsal, campeonato, liga
internet, infomática, email	família, adolescente, fevereiro	calendário, acadêmico, suspenso
acidente, tucuruí, ônibus	música, banda, rádio	

5.3.5 Modelagem de Tópicos na Região Sul

Na região Sul, conforme apresentado na Figura 40, o modelo BERTopic-BERTweet-BR apresentou um desempenho superior em comparação aos demais modelos, destacando-se pela maior quantidade de tópicos relacionados aos sentimentos positivos e negativos, como é apresentado na Tabela 3.

[2] Fonte: G1, disponível em: <https://g1.globo.com/pa/para/noticia/2024/05/29/policia-cientifica-realiza-pericia-em-onibus-do-ifpa-envolvido-em-acidente-com-4-mortes-em-tucurui.ghtml>

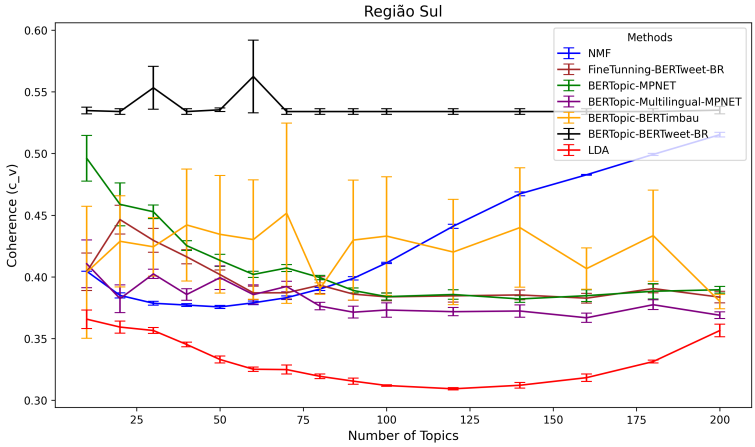


Figura 40 – Coerência: Região Sul.

Na região Sul, os tópicos foram gerados pelo modelo BERTopic-BERTweet-BR, uma vez que este apresentou o ponto de maior coerência, sendo gerado com 50 tópicos. Desta forma, conforme apresentado na Tabela 8, a análise evidencia uma predominância de discussões relacionadas à rotina dos estudantes e à preparação para exames. Esse padrão é observado em tópicos que contêm os termos *prova*, *estudando* e *fazer*, bem como no tópico com os termos *amanhã*, *prova* e *acordar*, além do tópico que inclui os termos *questões*, *prova* e *matemática*.

Tabela 8 – Distribuição dos tópicos na região Sul

Tópicos		
câmpus, estudantes, ensino	prova, estudando, fazer	feliz, matriculado, estudando
vontade, medo, passar	passei, primeiro, óbvio	engenharia, elétrica, eletrônica
episódio, canal, porta	salve, turma, química	saudáveis, conversas, garantidas
curso, técnico, ensino	gente, estudava, época	formatura, festa, indispensável
aulas, voltar, começam	fraco, forte, aluno	amanhã, prova, acordar
especialização, gratuito, inscrições	saudades, vontade, admitir	questões, prova, matemática
deus, graças, mente	expulso, acredite, caprichar	

Adicionalmente, há menções a eventos acadêmicos e cerimônias estudantis, destacadas em tópico que apresentam os termos *formatura*, *festa* e *indispensável*. Por fim, observa-se a relevância da comunicação institucional na divulgação de inscrições, evidenciada no tópico compostos pelos termos *especialização*, *gratuito* e *inscrições*.

6 Conclusão

Este estudo analisou diferentes técnicas de modelagem de tópicos aplicadas ao Twitter, com foco nas discussões sobre os Institutos Federais (IFs), considerando a distribuição regional das interações no Brasil e a segmentação por sentimento (positivo, negativo e neutro). Os dados foram coletados continuamente entre agosto de 2023 e agosto de 2024, resultando em um total de 66.000 tweets. Após o pré-processamento, adotou-se uma abordagem comparativa envolvendo sete modelos distintos, incluindo cinco variantes de modelos utilizando BERTopic e métodos tradicionais, como Non-Negative Matrix Factorization (NMF) e Latent Dirichlet Allocation (LDA).

Os resultados mostraram que os métodos de modelagem de tópicos baseados em redes neurais, especificamente o BERTopic, superaram as abordagens convencionais. Os modelos BERTopic-Multilingual-MPNET, BERTopic-MPNET e Fine-Tuning-BERTweet-BR se destacaram no conjunto de dados geral, enquanto os modelos BERTopic-BERTweet e BERTopic-BERTimbau foram mais eficazes nas coleções separadas por sentimentos. A análise das discussões no Twitter revelou tópicos relevantes em diferentes regiões do Brasil, bem como temas negativos/positivos nas segmentações de tópicos por sentimento.

Apesar dos avanços alcançados, algumas limitações foram identificadas ao longo do estudo. A modelagem de tópicos aplicada ao Twitter apresenta desafios inerentes, especialmente devido à informalidade da linguagem utilizada pelos usuários da plataforma, caracterizada pelo uso de gírias, erros gramaticais e abreviações. Embora técnicas baseadas em redes neurais permitam a representação desses termos, novos estudos são necessários para avaliar seu impacto na coerência dos tópicos.

Para pesquisas futuras, sugere-se a exploração de novas estratégias de Modelagem de Tópicos, como o *prompting* em modelos de linguagem de grande porte (LLM), visando aprimorar a identificação e descrição de tópicos sem a necessidade de ajustes manuais extensivos. Algumas direções promissoras incluem a exploração de *prompts* de LLMs para geração de tópicos, como recentemente proposto *TopicGPT* (WANG et al., 2023; PHAM et al., 2024). Além disso, o escopo da pesquisa pode ser ampliado para incluir um conjunto mais abrangente de universidades públicas brasileiras, possibilitando comparações entre diferentes instituições de ensino superior.

Outro avanço relevante seria o desenvolvimento de um sistema de monitoramento em tempo real, capaz de fornecer análises dinâmicas sobre os tópicos discutidos, permitindo um acompanhamento contínuo das tendências e a identificação precoce de temas emergentes. Tal aplicação poderia ter um impacto significativo na gestão da reputação institucional e na formulação de estratégias de comunicação.

Referências

ABDELRAZEK, A. et al. Topic modeling algorithms and applications: A survey. *Information Systems*, v. 112, p. 102131, 2023. ISSN 0306-4379. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0306437922001090>. Citado 2 vezes nas páginas 45 e 46.

ALAMSYAH, A. et al. Dynamic large scale data on twitter using sentiment analysis and topic modeling. In: *2018 6th International Conference on Information and Communication Technology (ICoICT)*. [S.l.: s.n.], 2018. p. 254–258. Citado 4 vezes nas páginas 13, 57, 58 e 62.

AMBROSINO, A. et al. What topic modeling could reveal about the evolution of economics*. *Journal of Economic Methodology*, Routledge, v. 25, n. 4, p. 329–348, 2018. Disponível em: <https://doi.org/10.1080/1350178X.2018.1529215>. Citado na página 45.

ARIFFIN, S. N. A. N.; TIUN, S. Rule-based text normalization for malay social media texts. *International Journal of Advanced Computer Science and Applications*, v. 11, 2020. Disponível em: <https://api.semanticscholar.org/CorpusID:229215969>. Citado na página 56.

AUBAID, A.; MISHRA, A. Text classification using word embedding in rule-based methodologies: A systematic mapping. v. 7, p. 902–914, 11 2018. Citado na página 23.

AZLAH, M. A. F. et al. Review on techniques for plant leaf classification and recognition. *Computers*, v. 8, n. 4, 2019. ISSN 2073-431X. Disponível em: <https://www.mdpi.com/2073-431X/8/4/77>. Citado 2 vezes nas páginas 5 e 19.

BA, J. L.; KIROS, J. R.; HINTON, G. E. *Layer Normalization*. 2016. Citado na página 39.

BLEI, D. M. Probabilistic topic models. *Commun. ACM*, Association for Computing Machinery, New York, NY, USA, v. 55, n. 4, p. 77–84, apr 2012. ISSN 0001-0782. Disponível em: <https://doi.org/10.1145/2133806.2133826>. Citado 3 vezes nas páginas 5, 48 e 49.

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 7 vezes nas páginas 5, 14, 47, 49, 50, 51 e 66.

CAMPELLO, R. J. G. B.; MOULAVI, D.; SANDER, J. Density-based clustering based on hierarchical density estimates. In: PEI, J. et al. (Ed.). *Advances in Knowledge Discovery and Data Mining*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013. p. 160–172. ISBN 978-3-642-37456-2. Citado na página 55.

CARMO, I. et al. Gerenciamento de dados de redes sociais com análise de redes e modelagem de tópicos. SBC, Porto Alegre, RS, Brasil, p. 64–70, 2023. ISSN 0000-0000. Disponível em: <https://sol.sbc.org.br/index.php/sbbd_estendido/article/view/25615>. Citado 3 vezes nas páginas 13, 57 e 58.

CARNEIRO, F. et al. Bertweet.br: a pre-trained language model for tweets in portuguese. *Neural Comput. Appl.*, Springer-Verlag, Berlin, Heidelberg, v. 37, n. 6, p. 4363–4385, dez. 2024. ISSN 0941-0643. Disponível em: <https://doi.org/10.1007/s00521-024-10711-3>. Citado 2 vezes nas páginas 66 e 68.

CASTRO, L. D.; FERRARI, D. Introdução à mineração de dados: Conceitos básicos, algoritmos e aplicações. *ISBN*, 2017. Citado 2 vezes nas páginas 5 e 63.

DEVLIN, J. et al. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. 2019. Citado 7 vezes nas páginas 5, 25, 40, 41, 42, 43 e 53.

DIENG, A. B.; RUIZ, F. J. R.; BLEI, D. M. Topic modeling in embedding spaces. *Transactions of the Association for Computational Linguistics*, v. 8, p. 439–453, 07 2020. ISSN 2307-387X. Disponível em: <https://doi.org/10.1162/tacl_a_00325>. Citado na página 70.

DIT, B. et al. Feature location in source code: a taxonomy and survey. *Journal of Software: Evolution and Process*, v. 25, n. 1, p. 53–95, 2013. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1002/smr.567>. Citado na página 46.

DOOGAN, C.; BUNTINE, W. Topic model or topic twaddle? re-evaluating semantic interpretability measures. Association for Computational Linguistics, Online, p. 3824–3848, jun. 2021. Disponível em: <https://aclanthology.org/2021.naacl-main.300>. Citado na página 47.

EGGER, R.; YU, J. Identifying hidden semantic structures in instagram data: A topic modelling comparison. *Tourism Review*, ahead-of-print, 10 2021. Citado na página 56.

FILHO, J. A. W. et al. The brWaC corpus: A new open resource for Brazilian Portuguese. In: CALZOLARI, N. et al. (Ed.). *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. Miyazaki, Japan: European Language Resources Association (ELRA), 2018. Disponível em: <https://aclanthology.org/L18-1686/>. Citado na página 75.

GARCIA, J. et al. Enhancing architectural recovery using concerns. p. 552–555, 2011. Citado na página 46.

GÉRON, A. *Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. 1st. ed. [S.l.]: O'Reilly Media, Inc., 2017. ISBN 1491962291. Citado 5 vezes nas páginas 5, 16, 17, 18 e 19.

GOLDBERG, Y.; HIRST, G. *Neural Network Methods in Natural Language Processing*. [S.l.]: Morgan & Claypool Publishers, 2017. (Synthesis Lectures on Human Language Technologies). ISBN 9781627052955. Citado na página 28.

GREFENSTETTE, G. Tokenization. In: _____. *Syntactic Wordclass Tagging*. Dordrecht: Springer Netherlands, 1999. p. 117–133. ISBN 978-94-015-9273-4. Disponível em: <https://doi.org/10.1007/978-94-015-9273-4_9>. Citado na página 28.

GRIFFITHS, T. L.; STEYVERS, M. Finding scientific topics. *Proceedings of the National Academy of Sciences*, v. 101, n. suppl_1, p. 5228–5235, 2004. Disponível em: <https://www.pnas.org/doi/abs/10.1073/pnas.0307752101>. Citado na página 48.

GROOTENDORST, M. Bertopic: Neural topic modeling with a class-based tf-idf procedure. 03 2022. Citado 6 vezes nas páginas 14, 16, 53, 56, 68 e 74.

HAYKIN, S. S. *Neural Networks and Learning Machines*. [S.l.]: Pearson, 2009. (Pearson International Edition). ISBN 9780131293762. Citado 2 vezes nas páginas 5 e 18.

HE, K. et al. *Deep Residual Learning for Image Recognition*. 2015. Citado na página 38.

HEIMERL, F.; GLEICHER, M. Interactive analysis of word vector embeddings. *Computer Graphics Forum*, v. 37, n. 3, p. 253–265, 2018. Citado na página 26.

HELLWIG, N. C. et al. Exploring twitter discourse with bertopic: topic modeling of tweets related to the major german parties during the 2021 german federal election. *International Journal of Speech Technology*, v. 27, n. 4, p. 901–921, December 2024. ISSN 1572-8110. Disponível em: <https://doi.org/10.1007/s10772-024-10142-4>. Citado 2 vezes nas páginas 77 e 80.

HEMMATI, H. et al. Prioritizing manual test cases in rapid release environments. *Software Testing, Verification and Reliability*, v. 27, n. 6, p. e1609, 2017. E1609 stvr.1609. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1002/stvr.1609>. Citado na página 46.

HEO, G. et al. Analyzing the field of bioinformatics with the multi-faceted topic modeling technique. *BMC Bioinformatics*, v. 18, 05 2017. Citado na página 46.

HINDLE, A. et al. Relating requirements to implementation via topic analysis: Do topics extracted from requirements make sense to managers and developers? p. 243–252, 2012. Citado na página 46.

HOANG, T.-A.; COHEN, W. W.; LIM, E.-P. On modeling community behaviors and sentiments in microblogging. p. 479–487, 2014. Disponível em: <https://epubs.siam.org/doi/abs/10.1137/1.9781611973440.55>. Citado 3 vezes nas páginas 13, 57 e 58.

HONG, S.; PARK, T.; CHOI, J. Analyzing research trends in university student experience based on topic modeling. *hong2020stainability*, v. 12, n. 9, 2020. ISSN 2071-1050. Disponível em: <https://www.mdpi.com/2071-1050/12/9/3570>. Citado 4 vezes nas páginas 13, 57, 59 e 60.

HOYLE, A. et al. Is automated topic model evaluation broken?: The incoherence of coherence. 2021. Citado na página 47.

JEONG, B.; YOON, J.; LEE, J.-M. Social media mining for product planning: A product opportunity mining approach based on topic modeling and sentiment analysis. *International Journal of Information Management*, v. 48, p. 280–290, 2019. ISSN 0268-4012. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0268401217302955>. Citado 2 vezes nas páginas 45 e 62.

JOACHIMS, T. A probabilistic analysis of the rocchio algorithm with tfidf for text categorization. In: . [S.l.: s.n.], 1996. p. 143–151. Citado na página 55.

JOULIN, A. et al. *Bag of Tricks for Efficient Text Classification*. 2016. Citado na página 24.

JR, S. B.; TAVARES, G. M.; KIDO, G. S. Artificial and natural topic detection in online social networks. *iSys - Brazilian Journal of Information Systems*, v. 10, n. 1, p. 80–98, Mar. 2017. Disponível em: <https://sol.sbc.org.br/journals/index.php/isys/article/view/329>. Citado na página 13.

KOLTCOV, S. et al. *Analysis and tuning of hierarchical topic models based on Renyi entropy approach*. 2021. Citado na página 47.

KUANG, D.; BRANTINGHAM, P.; BERTOZZI, A. Crime topic modeling. *Crime Science*, v. 6, 12 2017. Citado 2 vezes nas páginas 5 e 52.

KUANG, W.; LUO, N.; SUN, Z. Resource recommendation based on topic model for educational system. v. 2, p. 370–374, 2011. Citado 3 vezes nas páginas 13, 59 e 60.

LAUREATE, C. D. P.; BUNTINE, W.; LINGER, H. A systematic review of the use of topic models for short text social media analysis. *Artif. Intell. Rev.*, Kluwer Academic Publishers, USA, v. 56, n. 12, p. 14223–14255, may 2023. ISSN 0269-2821. Disponível em: <https://doi.org/10.1007/s10462-023-10471-x>. Citado na página 56.

LEE, D.; SEUNG, H. Learning the parts of objects by non-negative matrix factorization. *Nature*, v. 401, p. 788–91, 11 1999. Citado 2 vezes nas páginas 14 e 51.

LEE, D.; SEUNG, H. Algorithms for non-negative matrix factorization. *Adv. Neural Inform. Process. Syst.*, v. 13, 02 2001. Citado 2 vezes nas páginas 51 e 53.

LIU, L. et al. An overview of topic modeling and its current applications in bioinformatics. *SpringerPlus*, v. 5, 09 2016. Citado na página 46.

MA, G. Tweets classification with bert in the field of disaster management. In: *Workshop Department of Civil Engineering 2019*. [S.l.: s.n.], 2019. p. 1–15. Citado na página 43.

MCCULLOCH, W.; PITTS, W. A logical calculus of ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, v. 5, p. 127–147, 1943. Citado na página 17.

MCINNES, L.; HEALY, J.; MELVILLE, J. *UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction*. 2018. Citado na página 54.

MIKOLOV, T. et al. *Distributed Representations of Words and Phrases and their Compositionality*. 2013. Citado na página 24.

OZCIFT, A. et al. Advancing natural language processing (nlp) applications of morphologically rich languages with bidirectional encoder representations from transformers (bert): an empirical case study for turkish. *Automatika*, v. 62, p. 1–13, 05 2021. Citado 3 vezes nas páginas 5, 42 e 44.

PENNINGTON, J.; SOCHER, R.; MANNING, C. GloVe: Global vectors for word representation. In: MOSCHITTI, A.; PANG, B.; DAELEMANS, W. (Ed.). *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, 2014. p. 1532–1543. Disponível em: <https://aclanthology.org/D14-1162>. Citado na página 24.

PETERS, M. E. et al. *Deep contextualized word representations*. 2018. Citado na página 25.

PHAM, C. M. et al. *TopicGPT: A Prompt-based Topic Modeling Framework*. 2024. Disponível em: <https://arxiv.org/abs/2311.01449>. Citado na página 86.

PINTO, M. A. S. et al. Relacionando modelagem de tópicos e classificação de sentimentos para análise de mensagens do twitter durante a pandemia da covid-19. SBC, Porto Alegre, RS, Brasil, p. 61–64, 2020. ISSN 2596-1683. Disponível em: <https://sol.sbc.org.br/index.php/webmedia_estendido/article/view/13064>. Citado 2 vezes nas páginas 13 e 57.

QUEIROZ, M. M. A framework based on twitter and big data analytics to enhance sustainability performance. *Environmental Quality Management*, Wiley Online Library, v. 28, n. 1, p. 95–100, 2018. Citado na página 56.

RADFORD, A. et al. Improving language understanding by generative pre-training. In: . [s.n.], 2018. Disponível em: <https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf>. Citado na página 25.

REFAEILZADEH, P.; TANG, L.; LIU, H. Cross-validation. In: _____. *Encyclopedia of Database Systems*. Boston, MA: Springer US, 2009. p. 532–538. ISBN 978-0-387-39940-9. Disponível em: <https://doi.org/10.1007/978-0-387-39940-9_565>. Citado 2 vezes nas páginas 5 e 70.

REIMERS, N.; GUREVYCH, I. *Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks*. 2019. Citado na página 54.

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*. New York, NY, USA: Association for Computing Machinery, 2015. (WSDM '15), p. 399–408. ISBN 9781450333177. Disponível em: <https://doi.org/10.1145/2684822.2685324>. Citado na página 69.

ROSENBLATT, F. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, v. 65, n. 6, p. 386—408, November 1958. ISSN 0033-295X. Disponível em: <https://doi.org/10.1037/h0042519>. Citado na página 17.

RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. Learning representations by back-propagating errors. *nature*, Nature Publishing Group UK London, v. 323, n. 6088, p. 533–536, 1986. Citado na página 19.

SANANDRES, E.; ABELLO, R.; MADARIAGA, C. Topic modeling of twitter conversations: The case of the national university of colombia. In: IEZZI, D. F.; MAYAFFRE, D.; MISU15065003RACA, M. (Ed.). *Text Analytics*. Cham: Springer International Publishing, 2020. p. 241–251. ISBN 978-3-030-52680-1. Citado 3 vezes nas páginas 13, 59 e 60.

SIA, S.; DUH, K. Adaptive mixed component LDA for low resource topic modeling. Association for Computational Linguistics, Online, p. 2451–2469, abr. 2021. Disponível em: <https://aclanthology.org/2021.eacl-main.209>. Citado na página 46.

SOUZA, F.; NOGUEIRA, R.; LOTUFO, R. BERTimbau: pretrained BERT models for Brazilian Portuguese. In: *9th Brazilian Conference on Intelligent Systems, BRACIS, Rio Grande do Sul, Brazil, October 20-23 (to appear)*. [S.l.: s.n.], 2020. Citado na página 68.

SUN, L.; YIN, Y. Discovering themes and trends in transportation research using topic modeling. *Transportation Research Part C: Emerging Technologies*, v. 77, p. 49–66, 2017. ISSN 0968-090X. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0968090X17300207>. Citado na página 45.

TERRAGNI, S. et al. OCTIS: Comparing and optimizing topic models is simple! In: GKATZIA, D.; SEDDAH, D. (Ed.). *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*. Online: Association for Computational Linguistics, 2021. p. 263–270. Disponível em: <https://aclanthology.org/2021.eacl-demos.31>. Citado 2 vezes nas páginas 13 e 47.

VASWANI, A. et al. Attention is all you need. In: GUYON, I. et al. (Ed.). *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2017. v. 30. Disponível em: <https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf>. Citado 10 vezes nas páginas 5, 26, 27, 30, 31, 33, 36, 37, 38 e 40.

WANG, H. et al. *Prompting Large Language Models for Topic Modeling*. 2023. Disponível em: <https://arxiv.org/abs/2312.09693>. Citado na página 86.

WANG, Y. et al. From static to dynamic word representations: a survey. *International Journal of Machine Learning and Cybernetics*, v. 11, p. 1611–1630, 2020. Citado na página 29.

WEBBER, W.; MOFFAT, A.; ZOBEL, J. A similarity measure for indefinite rankings. *ACM Trans. Inf. Syst.*, Association for Computing Machinery, New York, NY, USA, v. 28, n. 4, nov. 2010. ISSN 1046-8188. Disponível em: <https://doi.org/10.1145/1852102.1852106>. Citado na página 71.

XIANG, B.; ZHOU, L. Improving Twitter sentiment analysis with topic-based mixture modeling and semi-supervised training. Association for Computational Linguistics, Baltimore, Maryland, p. 434–439, jun. 2014. Disponível em: <https://aclanthology.org/P14-2071>. Citado 3 vezes nas páginas 57, 58 e 62.