



**UNIVERSIDADE FEDERAL DO CEARÁ**  
**CENTRO DE TECNOLOGIA**  
**DEPARTAMENTO DE TELEINFORMÁTICA**  
**CURSO DE GRADUAÇÃO EM ENGENHARIA DE COMPUTAÇÃO**

**JOHN VASCONCELOS DOS SANTOS**

**CLASSIFICAÇÃO UNI E MULTIRRÓTULO DE SENTIMENTOS EM TEXTOS**  
**CURTOS**

**FORTALEZA**

**2025**

JOHN VASCONCELOS DOS SANTOS

CLASSIFICAÇÃO UNI E MULTIRRÓTULO DE SENTIMENTOS EM TEXTOS CURTOS

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Engenharia de Computação do Centro de Tecnologia da Universidade Federal do Ceará, como requisito parcial à obtenção do grau de bacharel em Engenharia de Computação.

Orientador: Prof. Dr. José Marques Soares

FORTALEZA

2025

Dados Internacionais de Catalogação na Publicação  
Universidade Federal do Ceará  
Sistema de Bibliotecas  
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

---

- S235c Santos, John Vasconcelos dos.  
Classificação uni e multirrótulo de sentimentos em textos curtos / John Vasconcelos dos Santos. – 2025.  
60 f. : il. color.
- Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Centro de Tecnologia,  
Curso de Engenharia de Computação, Fortaleza, 2025.  
Orientação: Prof. Dr. José Marques Soares.
1. Sentimentos. 2. Classificação. 3. Avaliação de emoções. 4. Classificação de sentimentos. 5. Inteligência artificial. I. Título.

CDD 621.39

---

JOHN VASCONCELOS DOS SANTOS

CLASSIFICAÇÃO UNI E MULTIRRÓTULO DE SENTIMENTOS EM TEXTOS CURTOS

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Engenharia de Computação do Centro de Tecnologia da Universidade Federal do Ceará, como requisito parcial à obtenção do grau de bacharel em Engenharia de Computação.

Aprovada em:

BANCA EXAMINADORA

---

Prof. Dr. José Marques Soares (Orientador)  
Universidade Federal do Ceará (UFC)

---

Prof. Dr. Michela Mulas  
Universidade Federal do Ceará (UFC)

---

Eng. Artur Rodrigues Rocha Neto  
Universidade Federal do Ceará (UFC)

À minha mãe, por sua obstinação de sempre, que me trouxe até aqui — ao ponto de escrever esta dedicatória — e que, tenho certeza, me levará ainda mais longe.

## AGRADECIMENTOS

Ao Prof. Dr. José Marques Soares pela orientação tranquila e técnica. E por ser igual coração de mãe e sempre aceitar mais um (orientando).

A GTi Engenharia Jr e sua salinha, a qual o número de horas que passei se convertem em dias, além de seus membros, sempre receptivos.

À Prof<sup>ª</sup>. Dra. Michela Mulas, pelos ensinamentos nas disciplinas de Introdução à Estatística e Inteligência Computacional Aplicada, que serviram de alicerce para este trabalho.

À Giulia Lacerda, Gisele Pontes e Jennifer Gadelha, pela amizade e pelo suporte fundamental na conclusão deste trabalho.

A Maycon Thiago, Victor Santos, Francisco de Assis, Teodoro Raulino e Victor Ehrich, cuja camaradagem todos esses anos ultrapassaram em muito o âmbito acadêmico.

Aos colegas que ajudaram no complemento das pesquisas respondendo meus formulários neste trabalho (e em tantos outros).

Aos meus irmãos que todos os dias me lembravam que eu tinha um TCC para entregar e sua cooperação para que eu conseguisse.

Aos demais colegas, amigos e professores que acompanharam minha jornada acadêmica e tornaram possível a apresentação desse trabalho de conclusão de curso.

Ao Doutorando em Engenharia Elétrica, Ednardo Moreira Rodrigues, e seu assistente, Alan Batista de Oliveira, aluno de graduação em Engenharia Elétrica, pela adequação do *template* utilizado neste trabalho para que o mesmo ficasse de acordo com as normas da biblioteca da Universidade Federal do Ceará (UFC).

“Se vi mais longe, foi por estar sobre ombros de gigantes.”

(Isaac Newton)

## RESUMO

Neste trabalho, investiga-se a eficácia de diferentes modelos de classificação na tarefa de identificar emoções em textos curtos extraídos de redes sociais, adotando abordagens uni e multirrótulo. Para isso, foi construído um corpus composto por 2209 comentários reais (*dataset* natural) e 1669 amostras geradas artificialmente por LLMs (*dataset* sintético), com o objetivo de mitigar o desbalanceamento de classes. Foram avaliados modelos tradicionais como *Support Vector Machine* (SVM), Perceptron Multicamadas, BERT e uma LLM (ChatGPT), considerando tanto classificações unirrótulo quanto multirrótulo. As amostras foram submetidas a tratamentos lexicais e tokenização, e os modelos treinados com diferentes composições dos dados. Os resultados indicam que o uso de dados sintéticos e técnicas de pré-processamento melhoraram significativamente a acurácia e generalização dos modelos, especialmente para classes minoritárias. O modelo BERT atingiu a melhor performance na abordagem unirrótulo (acurácia de 73,09%), enquanto a LLM obteve o melhor desempenho na classificação multirrótulo (Índice de Jaccard de 69,91%). Os achados destacam o potencial da combinação de dados sintéticos e pré-processamento na melhoria da classificação emocional em contextos com dados limitados e desbalanceados.

**Palavras-chave:** Classificação multirrótulo. Emoções. Dados sintéticos. Processamento de linguagem natural. Aprendizado de máquina.

## ABSTRACT

This work investigates the effectiveness of different classification models in the task of identifying emotions in short texts extracted from social media, adopting both single-label and multi-label approaches. To this end, a corpus was constructed comprising 2,209 real comments (natural dataset) and 1,669 samples artificially generated by LLMs (synthetic dataset), aiming to mitigate class imbalance. Traditional models such as Support Vector Machine (SVM), Multi-layer Perceptron, BERT, and a Large Language Model (ChatGPT) were evaluated, considering both single-label and multi-label classifications. The samples underwent lexical preprocessing and tokenization, and the models were trained with different data compositions. The results indicate that the use of synthetic data and preprocessing techniques significantly improved model accuracy and generalization, especially for minority classes. The BERT model achieved the best performance in the single-label approach (accuracy of 73.09%), while the LLM obtained the best performance in the multi-label classification (Jaccard Index of 69.91%). The findings highlight the potential of combining synthetic data and preprocessing techniques to improve emotion classification in contexts with limited and imbalanced data.

**Keywords:** Multi-label classification. Emotions. Synthetic data. Natural language processing. Machine learning.

## LISTA DE FIGURAS

Figura 1 – Ilustração de hiperplano canônico e separador . . . . .	20
Figura 2 – Exemplos de corpus linearmente separáveis (A) e não linearmente separáveis (B) . . . . .	21
Figura 3 – Representação típica das funções ReLU e GELU . . . . .	23
Figura 4 – Representação de uma rede neural recorrente . . . . .	24
Figura 5 – Processo de modelagem do classificador . . . . .	33
Figura 6 – Divisão das acurácias das quatro melhores e quatro piores frases avaliadas .	44
Figura 7 – Matriz de confusão do modelo sem tratamento do <i>dataset</i> . . . . .	48
Figura 8 – Matriz de confusão do modelo com tratamento do <i>dataset</i> . . . . .	48
Figura 9 – Matriz de confusão do SVM com tratamento do <i>dataset</i> . . . . .	50
Figura 10 – Matriz de confusão do SVM com tratamento do <i>dataset</i> e <i>dataset</i> sintético .	50
Figura 11 – Métricas de F1-Score para os algoritmos SVM, Perceptron e BERT . . . . .	51
Figura 12 – Métricas de Recall para os algoritmos SVM, Perceptron e BERT . . . . .	51
Figura 13 – Matriz de confusão do classificador SVM . . . . .	59
Figura 14 – Matriz de confusão do classificador SVM: corpus natural pré-processado . .	59
Figura 15 – Matriz de confusão do classificador SVM: corpus natural pré-processado + sintético . . . . .	60
Figura 16 – Matriz de confusão do classificador RNA . . . . .	60
Figura 17 – Matriz de confusão do classificador RNA: corpus natural pré-processado . .	61
Figura 18 – Matriz de confusão do classificador RNA: corpus natural pré-processado + sintético . . . . .	61

## LISTA DE TABELAS

Tabela 1 – Principais tipos de Kernel . . . . .	22
Tabela 2 – Exemplo de frases e suas classificações em um processo MLM abstraído . .	26
Tabela 3 – Exemplo de Amostras <i>IsNext</i> e <i>NotNext</i> . . . . .	27
Tabela 4 – Distribuição de Emoções 1 no corpus natural . . . . .	34
Tabela 5 – Distribuição de Emoções 1 no corpus sintético . . . . .	35
Tabela 6 – Tratamento lexicais aplicados no <i>dataset</i> . . . . .	36
Tabela 7 – Exemplos de ambiguidade ampliada após tratamentos lexicais . . . . .	37
Tabela 8 – Hiperparâmetros utilizados no SVM . . . . .	38
Tabela 9 – Hiperparâmetros utilizados no MPL . . . . .	39

## LISTA DE QUADROS

Quadro 1 – Métricas do modelo de <i>Support Vector Machine</i> (SVM) . . . . .	42
Quadro 2 – Métricas do modelo Perceptron multicamadas . . . . .	43
Quadro 3 – Métricas do modelo <i>Bidirectional Encoder Representations from Transfor-</i> <i>mers</i> (BERT) . . . . .	43
Quadro 4 – Métricas da <i>Large Language Model</i> (LLM) . . . . .	44
Quadro 5 – Métricas das avaliações via Formulário . . . . .	44
Quadro 6 – Métricas do modelo SVM Multirrótulo . . . . .	45
Quadro 7 – Métricas do modelo Perceptron multicamadas Multirrótulo . . . . .	46
Quadro 8 – Métricas da LLM Multirrótulo . . . . .	46
Quadro 9 – Acurácia dos modelos utilizados nesse trabalho . . . . .	46
Quadro 10 – Índice de Jaccard e <i>Hamming Loss</i> dos modelos multirrótulo utilizados nesse trabalho . . . . .	47
Quadro 11 – Métricas de generalização dos modelos . . . . .	49

## LISTA DE ABREVIATURAS E SIGLAS

API	<i>Application Programming Interface</i>
BERT	<i>Bidirectional Encoder Representations from Transformers</i>
CNN	<i>Convolutional Neural Networks</i>
GELU	<i>Gaussian Error Linear Unit</i>
IA	Inteligência Artificial
LLM	<i>Large Language Model</i>
MLM	<i>Masked Language Modeling</i>
MLP	<i>Multilayer Perceptron</i>
NSP	<i>Next Sentence Prediction</i>
PLN	Processamento de Linguagem Natural
ReLU	<i>Rectified Linear Unit</i>
RNA	Rede Neural Artificial
RNN	<i>Recurrent Neural Networks</i>
SMOTE	<i>Synthetic Minority Over-sampling Technique</i>
SVM	<i>Support Vector Machine</i>
TAE	Teoria do Aprendizado Estatístico
TEA	Transtorno do Espectro Autista
TF-IDF	<i>Term Frequency - Inverse Document Frequency</i>

## LISTA DE SÍMBOLOS

$\xi$	Variável de Folga
$\mathfrak{X}$	Espaço de Características ( <i>Feature space</i> )

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>17</b>
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>19</b>
<b>2.1</b>	<b>Support Vector Machines Lineares e Não lineares</b>	<b>19</b>
<i>2.1.1</i>	<i>SVMs com Margens rígidas</i>	<i>19</i>
<i>2.1.2</i>	<i>SVMs com Margens suaves</i>	<i>20</i>
<i>2.1.3</i>	<i>SVMs não lineares</i>	<i>21</i>
<b>2.2</b>	<b>Redes Neurais Artificiais</b>	<b>21</b>
<i>2.2.1</i>	<i>Neurônios: entradas e pesos</i>	<i>22</i>
<i>2.2.2</i>	<i>Função de ativação</i>	<i>22</i>
<i>2.2.3</i>	<i>Rede Neural Artificial (RNA)s multicamadas</i>	<i>22</i>
<i>2.2.4</i>	<i>RNAs recorrentes e feedforward</i>	<i>23</i>
<i>2.2.5</i>	<i>Perceptron Clássico e Multicamadas</i>	<i>24</i>
<i>2.2.6</i>	<i>Backpropagation</i>	<i>25</i>
<b>2.3</b>	<b>Transformers e BERT</b>	<b>25</b>
<i>2.3.1</i>	<i>MLM</i>	<i>25</i>
<i>2.3.2</i>	<i>NSP</i>	<i>26</i>
<b>2.4</b>	<b>Métricas de avaliação</b>	<b>27</b>
<i>2.4.1</i>	<i>Acurácia</i>	<i>27</i>
<i>2.4.2</i>	<i>Recall</i>	<i>28</i>
<i>2.4.3</i>	<i>Precisão</i>	<i>28</i>
<i>2.4.4</i>	<i>F1-score</i>	<i>28</i>
<i>2.4.5</i>	<i>Índice de Jaccard</i>	<i>28</i>
<i>2.4.6</i>	<i>Hamming loss</i>	<i>28</i>
<b>2.5</b>	<b>Teoria das Emoções Discretas de Ekman</b>	<b>29</b>
<b>3</b>	<b>TRABALHOS RELACIONADOS</b>	<b>31</b>
<b>3.1</b>	<b>Considerações finais</b>	<b>32</b>
<b>4</b>	<b>METODOLOGIA</b>	<b>33</b>
<b>4.1</b>	<b>Datasets</b>	<b>33</b>
<i>4.1.1</i>	<i>Dataset Natural</i>	<i>33</i>
<i>4.1.2</i>	<i>Dataset Sintético</i>	<i>34</i>

<b>4.2</b>	<b>Pré-processamento</b>	35
4.2.1	<i>Tratamentos lexicais</i>	36
4.2.2	<i>TF-IDF</i>	37
<b>4.3</b>	<b>Treinamento dos modelos</b>	38
4.3.1	<i>SVM</i>	38
4.3.2	<i>Multilayer Perceptron</i>	39
4.3.3	<i>BERT</i>	39
4.3.4	<i>LLMs</i>	40
4.3.5	<i>Avaliação Humana</i>	40
<b>4.4</b>	<b>Considerações finais</b>	40
<b>5</b>	<b>RESULTADOS</b>	42
<b>5.1</b>	<b>Resultados da classificação unirrótulo</b>	42
5.1.1	<i>Avaliação do SVM</i>	42
5.1.2	<i>Avaliação do modelo Perceptron Multicamadas</i>	43
5.1.3	<i>Avaliação do modelo BERT</i>	43
5.1.4	<i>Avaliação da LLM</i>	43
5.1.5	<i>Avaliação humana</i>	44
<b>5.2</b>	<b>Resultados da classificação multirrótulo</b>	45
5.2.1	<i>Avaliação do modelo SVM multirrótulo</i>	45
5.2.2	<i>Avaliação do modelo Perceptron Multicamadas multirrótulo</i>	45
5.2.3	<i>Avaliação da LLM utilizando as métricas de multirrótulo</i>	45
<b>5.3</b>	<b>Análise dos modelos unirrótulo</b>	46
<b>5.4</b>	<b>Análise dos modelos multirrótulo</b>	47
<b>5.5</b>	<b>Tratamentos Lexicais dos comentários</b>	47
<b>5.6</b>	<b>Dados sintéticos</b>	49
<b>6</b>	<b>CONCLUSÕES E TRABALHOS FUTUROS</b>	52
<b>6.1</b>	<b>Limitações deste trabalho e trabalhos futuros</b>	52
6.1.1	<i>Limitações</i>	52
6.1.2	<i>Trabalhos Futuros</i>	53
<b>6.2</b>	<b>Considerações finais e contribuições</b>	54
	<b>REFERÊNCIAS</b>	56
	<b>APÊNDICES</b>	59

<b>APÊNDICE A – Matrizes de confusão dos modelos unirrótulo . . . . .</b>	<b>59</b>
<b>ANEXOS . . . . .</b>	<b>59</b>

## 1 INTRODUÇÃO

Atualmente, a área de Processamento de Linguagem Natural (PLN) apresenta um crescimento significativo. O principal objetivo dessa área é processar a linguagem cotidiana dos seres humanos, identificando padrões e realizando operações que permitam compreender o conteúdo das mensagens. Esse entendimento pode, por exemplo, abranger a emoção que o autor da mensagem estava sentindo ou desejava transmitir. Como será explorado no Capítulo 2, classificadores baseados em inteligência artificial também são capazes de identificar padrões em mensagens e, com isso, realizar rotulações, mesmo que, em alguns casos, sem interpretar o conteúdo de forma contextual.

Mensagens com tons inapropriados repercutem rapidamente na internet, gerando polêmicas ou desencadeando ondas de reprovação pública. O engajamento em discursos de ódio tem aumentado, e há um consenso crescente de que as redes sociais devem ser capazes de identificar e inibir esse tipo de conteúdo. Mecanismos para detectar e remover discursos ofensivos em plataformas de compartilhamento de opiniões, como redes sociais, serviços de vídeo com seções de comentários, lojas virtuais com avaliações de produtos e sites de crítica, vêm sendo estudados (FORTUNA; NUNES, 2018).

Do ponto de vista do usuário, há uma busca crescente por compreender melhor as mensagens que são escritas e compartilhadas nas plataformas digitais. Algumas dessas plataformas oferecem mecanismos simples de feedback, como os sistemas de "gostei" (*like*) e "não gostei" (*dislike*), que auxiliam na identificação de níveis de aprovação ou rejeição de conteúdos. No entanto, fatores individuais podem interferir na interpretação emocional das mensagens. Por exemplo, estudos indicam que a empatia cognitiva, a capacidade de inferir os estados mentais e emocionais de outras pessoas, pode ser reduzida em indivíduos com Transtorno do Espectro Autista (TEA) (SILVA *et al.*, 2021).

Unindo esses contextos, classificadores baseados em Inteligência Artificial (IA) e PLN têm sido amplamente utilizados para a tarefa de identificação de emoções em textos publicados na internet. No Capítulo 3 é apresentada uma série de estudos que empregam classificadores com esse objetivo. Esses modelos são treinados a partir de grandes conjuntos de exemplos rotulados, denominados **conjuntos de treino**, para aprender a identificar corretamente as classes de novas amostras. Durante o processo de treinamento, o classificador é avaliado com base em um conjunto separado de dados previamente rotulados, chamado **conjunto de teste**, que permite medir sua capacidade de generalização. A junção do conjunto de treino com o conjunto

de teste constitui o **corpus** (ou *dataset*) utilizado no experimento.

No entanto, reunir uma grande quantidade de dados representativos para compor um corpus pode ser uma tarefa desafiadora. Emoções como **felicidade**, **tristeza** ou **raiva** são frequentemente expressas em redes sociais, enquanto outras, como **nojo** ou **medo**, aparecem com menor frequência, em razão do contexto e da linguagem típicos dessas plataformas. Diante disso, e com o avanço das LLMs, tornou-se cada vez mais comum o uso de **dados sintéticos**, que funciona como uma alternativa para suprir lacunas em corpora desbalanceados (PAVLYSHENKO; STASIUK, 2025).

Dados sintéticos são exemplos gerados artificialmente com o objetivo de simular o comportamento de dados reais. Esses dados podem ser produzidos por diferentes técnicas, como o método de sobreamostragem *Synthetic Minority Over-sampling Technique* (SMOTE) (CHAWLA *et al.*, 2002) ou a geração por meio de LLMs. A criação de dados sintéticos é especialmente relevante tanto para a construção de um novo corpus quanto para a **complementação de categorias minoritárias** em corpora existentes. No contexto da análise de sentimentos, por exemplo, a geração de amostras com o auxílio de LLMs permite o enriquecimento do corpus com exemplos de emoções sub-representadas, ampliando a diversidade e a qualidade do conjunto de dados.

Este trabalho teve como objetivo explorar diferentes tipos de classificadores no problema de classificação de emoções em textos curtos, sob as abordagens uni e multirrótulo. Foram treinados três modelos determinantes (SVM, Perceptron Multicamadas e BERT). Também foi realizada uma avaliação humana para estimar um “classificador humano médio” como base comparativa. Além disso, foi analisada a efetividade de grandes modelos de linguagem (LLMs) na resolução dessa tarefa. Técnicas complementares, como o pré-processamento dos dados e a geração de dados sintéticos, também foram investigadas, verificando-se seu impacto na melhoria do desempenho dos modelos.

O trabalho foi organizado da seguinte maneira: o Capítulo 2 apresenta os principais conceitos, algoritmos, teorias e técnicas utilizados. O Capítulo 3 descreve estudos anteriores relacionados ao tema, enquanto o Capítulo 4 detalha a jornada experimental realizada, destacando os parâmetros relevantes para a reprodutibilidade deste trabalho. No Capítulo 5, são apresentados os principais resultados obtidos, com ênfase nas métricas de desempenho dos algoritmos e nas variações observadas com e sem o uso das técnicas propostas. Por fim, o Capítulo 6 apresenta as conclusões e sugestões de aprimoramento para trabalhos futuros.

## 2 FUNDAMENTAÇÃO TEÓRICA

A seguir, são detalhados fundamentos dos principais algoritmos, técnicas e teorias empregadas neste trabalho.

### 2.1 Support Vector Machines Lineares e Não lineares

As máquinas de vetor de suporte (em inglês *support vector machines*) são técnicas de classificação que realizam, assim como outros algoritmos, procedimentos de maximização de margens. Técnicas desse tipo obtêm bons resultados de generalização, embasados pela Teoria do Aprendizado Estatístico (TAE), desenvolvida por Vapnik (1995).

#### 2.1.1 SVMs com Margens rígidas

Máquinas de vetor de suporte lineares funcionam em especial para dados linearmente separáveis, Boser *et al.* (1992). Mas alterações realizadas por Vapnik definem fronteiras lineares para dados gerais. Podemos definir os classificadores como Classificadores lineares e não lineares.

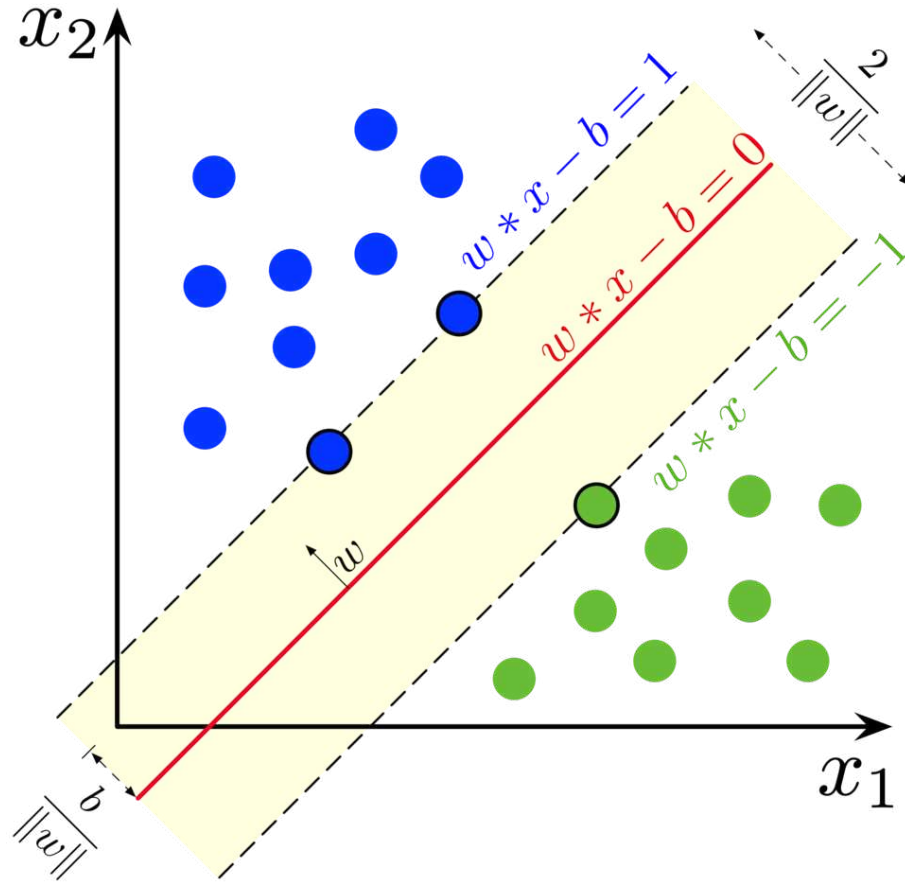
Classificadores lineares irão classificar corretamente um conjunto de resultados (ou rótulos)  $Y$ , separando um conjunto de variáveis  $x \in X$  de tamanho  $n$ , por um hiperplano. Uma ilustração dessa separação está disponível na Figura 1.

A equação do hiperplano que separa os dados é dada pela multiplicação escalar dos vetores  $x$  e  $w$ :  $h(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b$ , com  $\mathbf{w} \in \mathbf{X}$ , sendo um vetor perpendicular a esse mesmo hiperplano. Para a obtenção do classificador binário, que separam os dados linearmente, utiliza-se a função sinal:  $g(\mathbf{x}) = \text{sinal}(h(\mathbf{x})) = +1$  se  $\mathbf{w} \cdot \mathbf{x} + b > 0$  e  $-1$  se  $\mathbf{w} \cdot \mathbf{x} + b < 0$ .

Neste universo, existem infinitos hiperplanos que satisfazem essa condição. Assim, define-se o **hiperplano canônico** em relação a  $X$  como aquele em que os elementos mais próximos (vetores de suporte) ao hiperplano  $\mathbf{w} \cdot \mathbf{x} + b = 0$  satisfazem a equação  $|\mathbf{w} \cdot \mathbf{x} + b| = 1$ . Denotaremos essa condição por (1).

Agora, queremos maximizar a distância dos vetores de suporte para o hiperplano. Lembrando que temos um classificador binário de duas classes. Podemos utilizar a equação que define a distância de um ponto para o hiperplano  $\frac{|w \cdot x_0 + b|}{\|w\|}$ . Pela restrição (1), obtemos:  $\frac{1}{\|w\|}$ . Pretendemos maximizar a distância entre os dois vetores de suporte fixados, um para cada classe. Assim, podemos dobrar a distância e obtemos:  $\text{maximizar}(\frac{2}{\|w\|})$ . Podemos trocar esse problema

Figura 1 – Ilustração de hiperplano canônico e separador



Fonte: elaborado por Larhmam, CC BY-SA 4.0 <<https://creativecommons.org/licenses/by-sa/4.0/>>, via Wikimedia Commons.

de otimização pela minimização:  $\text{minimizar}(\frac{1}{2}\|w\|)$ . Este problema é resolvido sob a restrição de que, para cada ponto de treinamento  $(x_i, y_i)$  a condição  $y_i(w \cdot x + b) \geq 1$  seja satisfeita. Denominamos essa solução de margens rígidas.

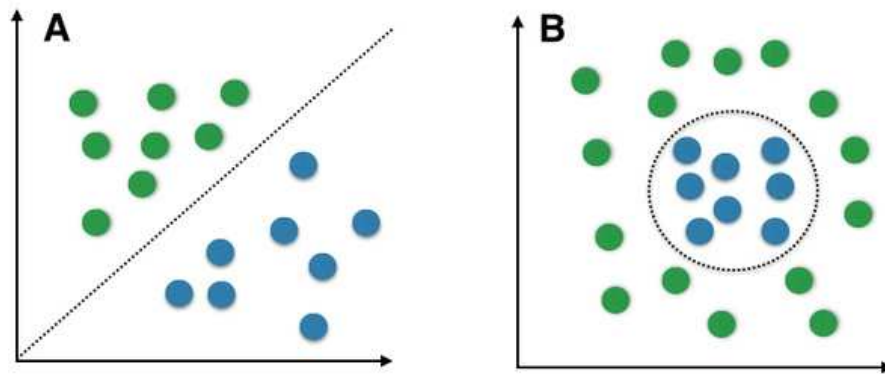
### 2.1.2 SVMs com Margens suaves

A definição de SVM de margens rígidas apresenta a limitação de classificar apenas amostras estritamente linearmente separáveis. Para contornar esse problema, e conseguir classificar amostras discrepantes (*outliers*), por exemplo, utiliza-se as variáveis de folga. Para qualquer amostra  $x_i$ , existe uma variável de folga  $\xi_i$ , que relaxa as restrições de otimização original, que se torna:  $y_i(w \cdot x_i + b) \geq 1 - \xi_i$  tal que  $\xi_i \geq 0$  para todo  $i = 1 \dots n$ . Assim, o problema de minimização é convertido em:  $\text{minimizar}(\frac{1}{2}\|w\| + C(\sum_{i=1}^n \xi_i))$ . O valor constante  $C$  é proporcional a penalização das violações contra a margem.

### 2.1.3 SVMs não lineares

Em alguns casos, não é possível realizar a classificação das amostras por meio de uma separação linear, mesmo com a utilização de margens suaves. Nessas situações, os SVMs lineares tornam-se inadequados para resolver o problema, sendo necessário o uso de SVMs não lineares. A Figura 2 apresenta dois exemplos. O Exemplo A, representa um corpus linearmente separável, enquanto o exemplo B exibe um corpus que não pode ser separado linearmente.

Figura 2 – Exemplos de corpus linearmente separáveis (A) e não linearmente separáveis (B)



Fonte: Sebastian Raschka, CC BY-SA 4.0 <<https://creativecommons.org/licenses/by-sa/4.0/>>, via Wikimedia Commons

SVMs não lineares conseguem lidar com problemas de dados linearmente inseparáveis mapeando as amostras do conjunto de treino em um espaço de maior dimensão, o espaço das características (*Features space*). Seja  $\phi = X \rightarrow \mathfrak{T}$  em que  $X$  é o espaço original e  $\mathfrak{T}$  é o espaço das características,  $\phi$  é o mapeamento de  $X$  em  $\mathfrak{T}$ . Segundo o **Teorema do Espaço de Características de Cover** (Cover (1965)),  $\mathfrak{T}$  tem altas chances probabilísticas de ser linearmente separável.

É possível encontrar o hiperplano utilizando a equação:  $h(x) = w \cdot \Phi(x) + b$  e em seguida calcular a otimização. Contudo, esse procedimento é computacionalmente oneroso. Opta-se pelo uso de funções denominadas kernels. Por definição, um kernel  $K$  é uma função que recebe dois pontos no espaço de entrada e calcula o produto desses pontos no espaço de características.  $K(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j)$ . O Quadro 1 apresenta os principais tipos de Kernel.

## 2.2 Redes Neurais Artificiais

As RNAs surgiram em estudos teóricos na década de 1940 com o objetivo de replicar computacionalmente o funcionamento do cérebro biológico. Foi somente na década de 1980

Tabela 1 – Principais tipos de Kernel

Tipo de Kernel	Função
Polinomial	$(\delta(x_i \cdot x_j) + K)^d$
RBF	$\exp(-\sigma \ x_i - x_j\ )$
Sigmoidal	$\tanh(\delta(x_i \cdot x_j) + K)$

Fonte: Faceli *et al.* (2024)

com o aumento da capacidade de processamento dos computadores e com novas técnicas de processamento paralelo que a área se expandiu vertiginosamente. A arquitetura das RNAs se baseia na fisiologia do cérebro humano, utilizando objetos como neurônios artificiais (simulando o neurônio biológico), terminais de entrada (dendritos), conexões entre neurônios de diferentes camadas (simulando as sinapses), dentre outros elementos.

### 2.2.1 Neurônios: entradas e pesos

Os neurônios são os elementos básicos das RNAs. Cada neurônio recebe entradas, realiza um processamento e gera uma saída. Um neurônio  $n$  com  $k$  terminais de entrada  $\mathbf{d}$  pode ser representado pelo vetor:  $n = [d_1, d_2, d_3 \dots d_k]$ . Podemos complementar essa definição com a atribuição de pesos ( $\mathbf{w}$ ) a cada entrada. Os pesos atribuem importância às conexões de entrada do neurônio. Assim, a representação do neurônio considerando os pesos torna-se:  $n = [w_1 d_1, w_2 d_2, w_3 d_3 \dots w_k d_k]$  A entrada total ( $\mathbf{u}$ ) do neurônio  $n$  pode ser expressa pela seguinte equação:  $u = \sum_{j=1}^k d_j w_j + b$ , onde  $b$  é o viés associado.

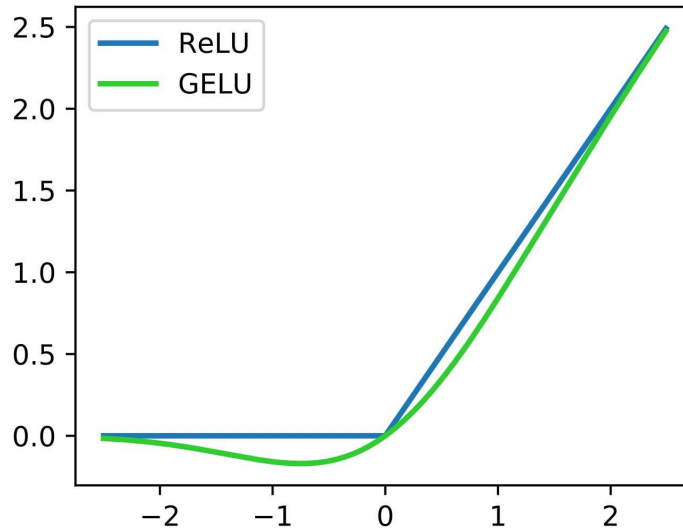
### 2.2.2 Função de ativação

Além dos valores de entrada, os neurônios definem uma saída a partir da *função de ativação*. A função de ativação atua sobre a entrada total:  $f(u) = y$ . Existem diversas funções de ativação típicas, como as funções lineares, limiares, sigmóides. Com a popularização das redes neurais profundas, as funções *Rectified Linear Unit* (ReLU) e *Gaussian Error Linear Unit* (GELU) se tornaram mais conhecidas e utilizadas. A Figura 3 apresenta as curvas típicas dessas funções.

### 2.2.3 RNAs multicamadas

Em geral as RNAs contam com múltiplos neurônios atuando em paralelo, recebendo entradas, calculando seus pesos e gerando saídas a partir das funções de ativação. Esse conjunto

Figura 3 – Representação típica das funções ReLU e GELU  
Nonlinearities



Fonte: Ringdongdang, CC BY-SA 4.0 <<https://creativecommons.org/licenses/by-sa/4.0/>>, via Wikimedia Commons

de neurônios que opera simultaneamente forma o que chamamos de camada da rede neural. Quando a rede é composta por várias camadas que se organizam em série, ela é denominada rede neural multicamada.

Em redes neurais desse tipo, as saídas dos neurônios da camada  $c_i$  tornam-se as entradas dos neurônios da camada subsequente,  $c_{i+1}$ . Existem diferentes tipos de padrão de conexão entre as camadas. Alguns padrões típicos de conexão entre camadas estão exemplificados:

- **Padrão Completamente conectado:** As saídas de todos os neurônios da camada  $c_i$  estão conectados em todos os neurônios da camada  $c_{i+1}$ .
- **Padrão Parcialmente conectado:** As saídas de alguns neurônios da camada  $c_i$  chegam nos neurônios da camada  $c_{i+1}$ .
- **Padrão Localmente conectado:** Saídas de neurônios específicos da camada  $c_i$  chegam nos neurônios da camada  $c_{i+1}$ .

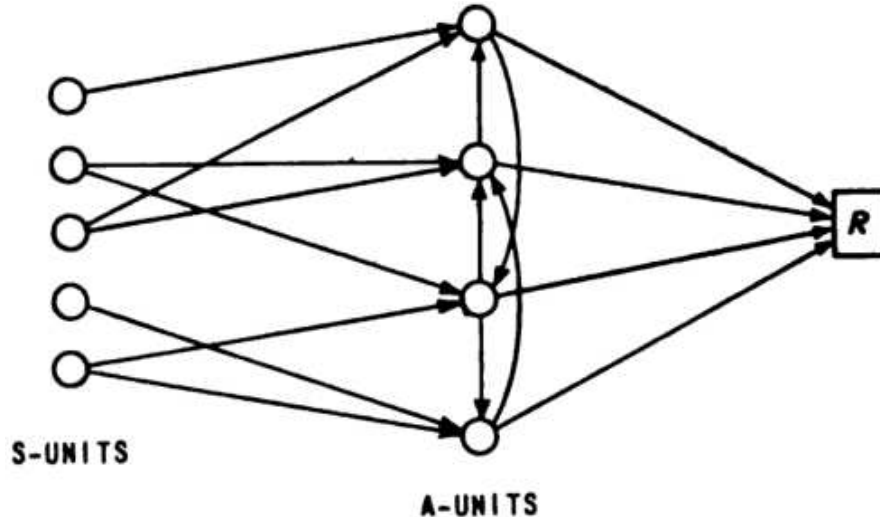
Em uma rede neural com  $i$  camadas, as camadas  $c_1 \dots c_{i-1}$  são denominadas **camadas intermediárias** ou **camadas ocultas** (*hidden layers*). Enquanto a camada  $c_0$  é denominada camada de entrada (*input layer*) e a camada  $c_i$  é denominada camada de saída (*output layer*).

#### 2.2.4 RNAs recorrentes e feedforward

O padrão de propagação de entradas e saídas nas redes neurais não é estritamente sequencial. Redes Neurais Retroalimentadas ou Redes Neurais Recorrentes se definem por redes neurais em que neurônios de uma camada posterior  $c_{i+1}$  alimenta neurônios de uma camada

anterior  $c_i$ ,  $n$  anterior:  $c_{i-n}$  ou até mesmo autoalimentação, onde neurônios da camada  $c_{i+1}$  alimentam neurônios da camada  $c_{i+1}$ . A Figura 4 apresenta uma rede neural recorrente clássica.

Figura 4 – Representação de uma rede neural recorrente



Fonte: Rosenblat (1961)

Em contrapartida as redes neurais recorrentes, existem as redes neurais sem retropropagação, chamadas de RNAs de *Feedforward*.

### 2.2.5 Perceptron Clássico e Multicamadas

A rede Perceptron, ou Perceptron de unicamada (Rosenblatt (1958)), é um modelo clássico de RNA que consiste em apenas uma única camada de neurônios. Utiliza uma função de ativação do tipo limiar (também conhecida como degrau unitário), definida pela seguinte equação:  $Y = 1$  se  $w \cdot x + b \geq 0$  ou  $Y = 0$ , caso contrário. Rosenblatt demonstrou que, sempre que um problema for linearmente separável, o Perceptron é capaz de encontrar uma solução de classificação adequada.

Uma extensão mais robusta desse modelo é o Perceptron Multicamadas (*Multilayer Perceptron* (MLP)), que adota uma arquitetura com múltiplas camadas. Essa configuração permite à rede resolver problemas mais complexos, que não podem ser resolvidos por separação linear. Cada camada adicional contribui para o aumento da capacidade de representação do modelo. Além disso, redes MLP empregam funções de ativação mais sofisticadas e variadas entre as camadas, como ReLU, sigmoide e tangente hiperbólica, o que amplia ainda mais sua flexibilidade na modelagem de padrões não lineares.

### 2.2.6 Backpropagation

Os modelos de redes neurais multicamadas tornaram-se viáveis após a introdução do algoritmo de retropropagação do erro (*backpropagation*), proposto por Rumelhart *et al.* (1986), que possibilitou o treinamento eficiente dessas arquiteturas mais profundas. O *backpropagation* pode ser dividido em duas fases, o *forward pass* e o *backward pass*.

A fase inicial, o *forward pass*, atua a partir da primeira camada intermediária da rede, onde os neurônios processam suas entradas por meio das funções de ativação, gerando saídas que são repassadas para as camadas seguintes. Esse processo continua até a camada de saída, que produz o resultado final da rede.

A fase de *backward* consiste na utilização do erro obtido para o cálculo de pesos das entradas dos neurônios ao longo da rede (camadas intermediárias). A seguinte equação descreve como são calculados os pesos nessa fase:  $w^{jl}(t+1) = w^{jl}(t) + \eta x^j \delta^l$ .  $w^{jl}$  representa o peso atribuído ao  $j$ -ésimo atributo de um neurônio  $l$  enquanto  $\delta^l$  representa o erro associado a esse  $l$ -ésimo neurônio e  $x^j$  representa a  $j$ -ésima entrada recebida por esse mesmo neurônio. O *backpropagation* estima o erro de ativação das camadas intermediárias por meio da soma dos erros do neurônio da camada posterior.

## 2.3 Transformers e BERT

A arquitetura Transformadores (ou *Transformers*) é uma abordagem diferente das *Recurrent Neural Networks* (RNN)s ou *Convolutional Neural Networks* (CNN)s, baseada **exclusivamente em atenção**. Foi proposta por Vaswani *et al.* (2017), sendo uma abordagem sequencial, focada em contexto. Atua com dois principais pontos: o **encoder** e o **decoder**.

Dentro do estudo de Transformadores, Devlin *et al.* (2018) proporam um método pré-treinado, o BERT. Esse modelo funciona com dois pilares essenciais, apoiados no **encoder**: *Masked Language Modeling* (MLM) e o *Next Sentence Prediction* (NSP).

### 2.3.1 MLM

O MLM apoia-se em uma função *softmax* MLMH (ou *MLM Head*) para resolver o seguinte problema: dentro do corpus de treinamento, 15% das palavras são substituídas pelo termo *[mask]*, termos aleatórios ou deixadas inalteradas. Utilizando contexto bilateral, a função *softmax* deve determinar a palavra correta que ocupava o espaço substituído, ou então declarar

que a palavra contida está correta para o contexto encontrado.

O MLM é uma tarefa executada no pré-treinamento do modelo BERT. Utiliza uma função softmax no componente conhecido como *MLM Head* para resolver a seguinte tarefa: dentro do corpus de treinamento, aproximadamente 15% das palavras são selecionadas aleatoriamente e substituídas por tokens especiais, como por exemplo o token *[MASK]*, por outras palavras aleatórias ou, em alguns casos, permanecem inalteradas. A partir do contexto bidirecional (considerando tanto o que vem antes quanto o que vem depois da palavra mascarada), a função softmax tem como objetivo prever qual era a palavra original substituída. Nos casos em que a palavra não foi alterada, o modelo deve ser capaz de reconhecer que ela está adequada ao contexto.

A Tabela 2 apresenta um exemplo abstraído do processo de MLM em linguagem natural. São exibidas frases com palavras substituídas por tokens (como *[MASK]*) e suas respectivas classificações corretas, ou seja, as palavras que o modelo deve prever com base no contexto.

Tabela 2 – Exemplo de frases e suas classificações em um processo MLM abstraído

Amostra	Classificação
O reino <b>[MASK]</b> engloba organismos unicelulares procariontes, como bactérias e arqueas.	Monera
<b>Cachorro</b> é um ramo da matemática que lida com a coleta e interpretação de dados.	Estatística
Lionel Messi é um futebolista argentino que atua como atacante, atualmente no <b>Inter Miami</b> .	Inter Miami

Fonte: elaborado pelo autor (2025)

### 2.3.2 NSP

O NSP é a segunda tarefa do pré-treinamento do modelo BERT. Seu objetivo é ensinar o modelo a determinar se uma frase é a continuação semântica de outra (*IsNext*) ou não (*NotNext*). Durante esse processo, o modelo recebe pares de frases. Em metade dos casos, a **Frase B** é a continuação lógica da **Frase A**, seguindo a sequência natural de um texto. Na outra metade, a **Frase B** é retirada de um ponto aleatório do corpus, não tendo relação direta com a **Frase A**.

A partir desses exemplos, o modelo aprende a identificar relações de continuidade semântica entre frases, utilizando uma camada de classificação com função softmax para prever se o par de frases deve ser rotulado como *IsNext* ou *NotNext*.

A Tabela 3 apresenta exemplos de pares de frases associados a cada um desses

rótulos.

Tabela 3 – Exemplo de Amostras *IsNext* e *NotNext*

	Frase A	Frase B	Rótulo
	Amanhã tenho aula!	A aula vai ser de ICA	IsNext
	Amanhã tenho aula!	Por isso, o nome do gato é Batata	NotNext
	Quando filhote comia as fritas do lanche	A aula vai ser ICA	NotNext
	Quando filhote comia as fritas do lanche	Por isso, o nome do gato é Batata	IsNext

Fonte: elaborado pelo autor (2025)

## 2.4 Métricas de avaliação

Na avaliação de modelos de classificação, utilizam-se diversas métricas para estimar a qualidade dos algoritmos. Muitas dessas métricas derivam da **matriz de confusão**, que se baseia nas definições de **verdadeiros positivos**, **verdadeiros negativos**, **falsos positivos** e **falsos negativos**. A seguir, apresentamos o significado de cada um desses termos:

- **Verdadeiro positivo (VP):** o modelo classificou corretamente uma amostra como pertencente a uma determinada classe. *Exemplo:* o modelo recebe a imagem de um gato e corretamente a classifica como pertencente à classe "felinos".
- **Verdadeiro negativo (VN):** o modelo classificou corretamente uma amostra como **não pertencente** a uma determinada classe. *Exemplo:* o modelo recebe a imagem de um cão e corretamente a classifica como "não-felino".
- **Falso Positivo (FP):** o modelo classificou **incorretamente** uma amostra como pertencente a uma classe. *Exemplo:* o modelo recebe a imagem de um leão-marinho e incorretamente a classifica como "felino".
- **Falso Negativo (FN):** o modelo classificou incorretamente uma amostra como não pertencente a uma classe. *Exemplo:* o modelo recebe a imagem de uma jaguatirica (um felino) e incorretamente a classifica como "não-felino".

Com base nesses conceitos, podemos definir as principais métricas de avaliação utilizadas neste trabalho:

### 2.4.1 Acurácia

A acurácia é a métrica mais simples e representa a proporção de previsões corretas em relação ao total de previsões feitas. Ela indica o desempenho geral do modelo.

$$acuracia = \frac{VP+VN}{VP+VN+FP+FN}$$

### 2.4.2 Recall

O recall, também chamado de sensibilidade, mede a capacidade do modelo em identificar corretamente todas as amostras positivas. É especialmente útil para avaliar o comportamento do modelo em relação a classes minoritárias.

$$recall = \frac{VP}{VP+FN}.$$

### 2.4.3 Precisão

A precisão mede a proporção de amostras classificadas como positivas que, de fato, pertencem à classe positiva. É uma métrica importante quando os falsos positivos têm alto custo.

$$precisao = \frac{VP}{VP+FP}.$$

### 2.4.4 F1-score

O F1-score é a média harmônica entre precisão e recall. Ele é útil quando se busca um equilíbrio entre esses dois valores, especialmente em contextos com classes desbalanceadas, onde a acurácia isoladamente pode ser enganosa.

$$F1 = 2 \cdot \frac{precisao \cdot recall}{precisao + recall}$$

### 2.4.5 Índice de Jaccard

Índice de Jaccard é uma métrica estatística utilizada para avaliar classificadores multirrótulo. Consiste na razão de dois conjuntos:  $J(A,B) = \frac{|A \cap B|}{|A \cup B|}$ . Em um cenário com um número L de rótulos e N amostras, pode ser descrito por:

$$I_j = \frac{1}{N} \cdot \sum_{i=1}^N \frac{f(p_i, r_i)}{g(p_i, r_i)}$$
 em que  $f(p_i, r_i)$  retorna a intersecção entre o i-ésimo vetor de rótulos preditos ( $p_i$ ) e o i-ésimo vetor de rótulos reais ( $r_i$ ). Enquanto,  $g(p_i, r_i)$  retorna o valor da união dos vetores  $p_i$  e  $r_i$ .

### 2.4.6 Hamming loss

Perda Hamming é outra métrica utilizada em *Machine Learning* para avaliar classificadores multirrótulo. Mede a quantidade rótulos que foram preditos incorretamente em relação a um número L de rótulos em N amostras.

$$H_l = \frac{1}{N \cdot L} \cdot \sum_{i=0}^n \sum_{j=0}^l f(k_1, k_2)$$

Em que  $f(k_1, k_2)$  é uma função que compara o  $j$ -ésimo rótulo predito  $r_{ij}$  da  $i$ -ésima amostra com o  $j$ -ésimo rótulo real  $R_{ij}$  dessa mesma amostra, retornando 0 caso sejam iguais e 1 caso sejam diferentes.

## 2.5 Teoria das Emoções Discretas de Ekman

Muito antes da aplicação de PLNs para a análise de sentimentos ou do uso de IAs para a classificação automática de emoções, o ser humano já buscava compreendê-las. Em *Retórica* (século IV a.C.), Aristóteles abordou o papel dos sentimentos (*pathos*) no processo de persuasão, procurando defini-los e entender as razões de suas manifestações (ARISTÓTELES, 2017). Após Aristóteles, diversos pensadores ofereceram novas perspectivas sobre o tema, ampliando o campo de estudo das emoções (DAMASIO, 1994; FRIJDA, 1986).

No final da década de 1960, Paul Ekman publicou uma das teses mais influentes na área da psicologia. Com base em pesquisas realizadas com tribos isoladas, Ekman propôs que algumas emoções são **primárias** e **universais**, ou seja, não são moldadas pela cultura nem por fatores ambientais. A principal tese de Ekman é que essas emoções podem ser distinguidas por características fisiológicas, como as expressões faciais. Essa proposta ficou conhecida como a **teoria das emoções discretas** (ou básicas).

Segundo Ekman, emoções como raiva, felicidade e tristeza são acompanhadas de expressões faciais reconhecíveis em todas as culturas, além de apresentarem características comportamentais e fisiológicas distintas. Essa definição contrasta com outras teorias que sugerem que as emoções diferem apenas em poucas dimensões (EKMAN, 1972).

Com base em suas pesquisas, Ekman identificou um conjunto de seis emoções primárias, caracterizadas por expressões faciais universais e respostas fisiológicas distintas: **felicidade, tristeza, raiva, medo, surpresa e nojo**. Em trabalhos posteriores, o autor chegou a considerar a possibilidade de que outras emoções também pudessem ser incluídas nesse grupo, desde que apresentassem características similares às das seis originais.

Em parte devido à sua simplicidade e objetividade, a teoria das emoções discretas de Ekman tem sido amplamente adotada em estudos de classificação de emoções com o uso de IAs e técnicas de PLN. Uma análise sistemática apresentada por (KHARE *et al.*, 2024), que abrange pesquisas realizadas entre 2014 e 2023, investigou diferentes abordagens de reconhecimento de emoções, utilizando meios como **eletrocardiograma** (ECG), **eletroencefalograma**

(EEG), **resposta galvânica da pele (GSR)**, **rastreamento ocular (ET)**, além da classificação de **expressões faciais e sinais físicos**.

Segundo o estudo, a teoria das emoções discretas foi a mais utilizada nas pesquisas baseadas em rastreamento ocular, expressões faciais, sinais físicos e EEG. Já nos trabalhos que utilizaram ECG e GSR, ela apareceu como a segunda abordagem mais recorrente. Tais dados reforçam a relevância da teoria de Ekman não apenas em contextos fisiológicos, mas também em aplicações computacionais e cognitivas. Abrimos o capítulo seguinte explorando o uso da teoria das emoções discretas Ekman em classificação de emoções baseados em texto na literatura.

### 3 TRABALHOS RELACIONADOS

Bhat (2024) apresentou o corpus *SmallEnglishEmotions*, um conjunto composto por 6.372 comentários curtos em inglês, cujos rótulos seguem a teoria de Ekman, sendo classificados como: **felicidade**, **raiva**, **tristeza**, **medo** e **outro**. Por sua vez, Ashraf *et al.* (2022) apresentaram um *dataset* no idioma urdu, do sul da Ásia, que abrange todas as seis emoções básicas propostas por Ekman. Já Alm *et al.* (2005) propuseram um classificador de emoções para sentenças extraídas de histórias infantis, com base nas emoções básicas de Ekman, mas com o acréscimo das categorias **surpresa positiva** e **surpresa negativa**. Pasupa e Seneewong Na Ayutthaya (2019) também trabalharam com classificadores de emoções em histórias infantis, testando modelos como *Convolutional Neural Network* (CNN), *Long Short-Term Memory* (LSTM) e *Bidirectional LSTM* (Bi-LSTM) para a classificação de emoções em textos no idioma tailandês, ainda no escopo de emoções de Ekman. Por fim, Filipe *et al.* (2020) exploraram técnicas de adaptação para o uso de datasets em inglês no treinamento de classificadores de emoções em português, utilizando o conjunto de emoções proposto por Ekman.

Sob a ótica de teorias de emoções distintas da proposta por Ekman, outros trabalhos têm sido apresentados na literatura. Kang *et al.* (2023) apresentam uma abordagem de aprendizado de máquina para a classificação multirrótulo de emoções em textos em corpora com viés acentuado, além de propor soluções para evitar o problema de sobreajuste (*overfitting*) em relação à emoção neutra. Um dos pontos relevantes do trabalho é a sugestão de métodos menos onerosos para a composição de corpora, com foco na maximização da eficiência dos classificadores humanos durante o processo de rotulagem.

Liu *et al.* (2023) concentraram-se no aprimoramento da classificação de emoções em textos curtos, especialmente em *tweets*, por meio de um método multirrótulo de aprendizado de máquina. Os autores destacaram a limitação de abordagens tradicionais para identificar múltiplas emoções expressas simultaneamente em um mesmo texto. Para contornar essa limitação, propõem um algoritmo denominado MLkNN modificado, que considera não apenas características internas à frase, mas também informações contextuais, como frases adjacentes e o texto completo, além das correlações entre os rótulos emocionais. A pesquisa demonstrou que a versão aprimorada do L-MLkNN supera outros métodos em desempenho geral, com destaque para a taxa de recall, indicando maior precisão na classificação entre múltiplas classes.

Qin *et al.* (2023) propõem uma abordagem para análise de sentimentos em microblogs, com foco no contexto da pandemia de COVID-19. O estudo utilizou um modelo do tipo BERT

(ALBERT), combinado com redes neurais TCN e BiGRU, para extração de características, que são posteriormente processadas por um classificador Softmax. Os resultados indicaram valores promissores de recall e F1-score, evidenciando a eficácia do uso combinado dessas arquiteturas.

Já Kansaon *et al.* (2019) e Hammes e Freitas (2021) focam na classificação de emoções em textos escritos em português. Kansaon *et al.* (2019) realizaram uma análise comparativa entre diversos algoritmos de classificação aplicados a um corpus de tweets em português. Foram avaliados mais de dez modelos, incluindo árvores de decisão e classificadores baseados em métodos bayesianos, sendo estes os que apresentaram melhor desempenho. Por sua vez, Hammes e Freitas (2021) trabalharam na otimização dos modelos BERTimbau-base e BERTimbau-large para a classificação de 27 tipos de emoção presentes no corpus *GoEmotions*, previamente traduzido para o português com o auxílio de ferramentas de tradução automática.

A literatura também tem explorado a atuação da PLN e de modelos de IA na classificação de discursos de ódio. D'Sa *et al.* (2022) investigaram o uso de técnicas de otimização para aprimorar a detecção de discurso de ódio por meio de LLMs. Já Liang (2021) realizou uma análise comparativa entre LLMs e algoritmos clássicos, como o SVM e o **XGBoost**, na tarefa de classificação de discurso de ódio, alcançando resultados promissores, com acurácia e recall superiores a 80%. Além disso, Silva e Freitas (2022) estudou a classificação de discursos de ódio utilizando o modelo **BERTimbau**, empregando três corpora em língua portuguesa, o que reforça a aplicabilidade de modelos linguísticos adaptados ao idioma na detecção de mensagens ofensivas.

### 3.1 Considerações finais

Diversos estudos propuseram analisar a classificação de comentários em diferentes tipos de rótulos emocionais, baseando-se em distintas teorias das emoções. Alguns desses trabalhos utilizaram bases de dados em português. Este estudo concentrou-se em explorar modelos pertencentes a paradigmas distintos de IA, como o SVM, o Perceptron multicamadas e o BERT. Também foi analisado como alterações na sintaxe dos dados influenciam a classificação, bem como o impacto da adição de dados sintéticos a bases de dados naturais em português. Por fim, contribuímos com uma análise da classificação multirrótulo de emoções nesse idioma, um campo ainda pouco explorado.

## 4 METODOLOGIA

Neste trabalho, foram classificadas emoções com base em um dataset de comentários curtos de redes sociais usando quatro abordagens: classificações com SVM, Perceptron multicamadas, BERT e LLMs. Essas abordagens são descritas com mais detalhes nas Sessões 4.3.1 a 4.3.5 e fundamentadas teoricamente no Capítulo 2. Os *datasets* utilizados estão descritos na sessão seguinte. Também exploramos o pré-processamento dos dados dos *datasets* na Sessão 4.2. Outrossim, foi realizada uma pesquisa, onde os participantes avaliaram amostras do *dataset*. Por fim, as métricas de avaliação estão descritas no Capítulo 5, traçando uma comparação dos métodos de classificação e a classificação humana, obtida por meio de aplicação de formulários. O processo de modelagem dos classificadores esta descrito na Figura 5.

Figura 5 – Processo de modelagem do classificador



Fonte: elaborado pelo autor (2025).

### 4.1 Datasets

Os Corpus ou *datasets* utilizados neste trabalho se dividem em dois, o *dataset* natural (composto por comentários reais de diversas redes sociais) e o *dataset* sintético (composto de dados simulados gerados por LLMs). Nas subseções seguintes são apresentadas as metodologias empregadas na concepção de ambos os *datasets*, além de mais detalhes. Os corpus estão disponíveis no **repositório**.

#### 4.1.1 Dataset Natural

O corpus natural é composto por 2209 amostras, coletadas manualmente em redes sociais, sendo 1142 do X (antigo Twitter), 445 do Instagram, 270 do YouTube Shorts, 241 do Reddit, 92 do Threads e 19 de outras plataformas.

Para a coleta dos dados, foram criadas novas contas em cada uma dessas redes, e as buscas foram realizadas utilizando o filtro "mais recente", sempre que disponível, priorizando exclusivamente publicações em português. No X, por exemplo, o uso do comando *lang:pt* permitiu localizar postagens de qualquer natureza escritas em língua portuguesa. Durante o

processo de seleção, foram descartados conteúdos promocionais, mensagens de spam e postagens compostas exclusivamente por imagens ou vídeos. Cada amostra contém cinco características (*features*): Comentário, Fonte, Emoção 1, Emoção 2 e Aspecto.

Neste contexto, a variável Comentário é utilizada como variável preditora, enquanto as variáveis Emoção 1, Emoção 2 e Aspecto são os rótulos a serem classificados. Todas essas *features* foram rotuladas manualmente para possibilitar o treinamento de modelos supervisionados.

As emoções foram categorizadas com base nas emoções básicas de Ekman: **felicidade, tristeza, raiva, surpresa, medo e nojo** (com a adição da emoção **neutra**). Além disso, a variável Emoção 2 pode, em alguns casos, estar vazia. A Tabela 4 apresenta a distribuição das emoções no campo Emoção 1 do corpus natural.

Tabela 4 – Distribuição de Emoções 1 no corpus natural

Rótulo	Quantidade presente no <i>dataset</i>
Felicidade	624
Raiva	662
Tristeza	326
Surpresa	252
Neutro	182
Medo	108
Nojo	55

Fonte: elaborado pelo autor (2025).

Já a variável Aspecto refere-se à análise do conteúdo dos comentários e foi rotulada manualmente em quatro categorias: positivo, negativo, ambíguo ou nulo. A variável Aspecto segue a seguinte distribuição: (negativo - 1164, positivo - 624, nulo - 349, ambíguo - 72).

#### 4.1.2 Dataset Sintético

Ao analisar o *dataset* natural, observa-se um desbalanceamento de classes: amostras rotuladas como **raiva**, por exemplo, são 12 vezes mais frequentes do que amostras rotuladas como **nojo**. Esse desbalanceamento pode prejudicar o desempenho de modelos clássicos, favorecendo o surgimento de problemas como o *overfitting*. Existem diversas estratégias para mitigar esse problema, entre elas a geração de dados artificiais.

A geração de dados sintéticos pode ser realizada por diferentes técnicas, como o SMOTE. A abordagem utilizada neste trabalho é a geração de amostras com o auxílio de LLMs. Para isso, foi criado um *dataset* sintético a partir de diversas LLMs gratuitas, resultando em 1669 novas amostras.

Cada amostra gerada contém cinco características (*features*): Comentário, Plataforma, Emoção 1, Emoção 2 e Aspecto. Assim como no *dataset* natural, o Comentário é a variável preditora, enquanto Emoção 1, Emoção 2 e Aspecto são os rótulos a serem classificados. Todas as *features* foram elaboradas e rotuladas pelas LLMs.

O corpus sintético foi produzido após a construção do corpus natural, com o objetivo de priorizar a geração de amostras pertencentes a classes minoritárias. A distribuição das emoções no campo Emoção 1 pode ser consultada na Tabela 5.

Tabela 5 – Distribuição de Emoções 1 no corpus sintético

Rótulo	Quantidade presente no <i>dataset</i>
Felicidade	147
Raiva	69
Tristeza	302
Surpresa	333
Neutro	37
Medo	384
Nojo	397

Fonte: elaborado pelo autor (2025).

A variável Aspecto segue a seguinte distribuição: (negativo - 1205, positivo - 171, nulo - 261, ambíguo - 30). Enquanto a variável Plataforma, que apresenta os LLMs que geraram os dados, está distribuída da seguinte maneira: (ChatGPT 4.0 - 541, X Grook - 331, Gemini 2.0 Flash - 287, Microsoft Copilot -186, Deepseek - 180, Meta AI - 134, Gemini 1.5 Flash - 10)

## 4.2 Pré-processamento

O tratamento dos dados brutos provenientes de ambos os *datasets* foi realizado antes do treinamento dos modelos. Essa etapa teve como objetivos principais: (i) adaptar as entradas (*features*) ao formato exigido pelos algoritmos, uma vez que alguns modelos não são capazes de processar diretamente dados textuais; e (ii) aprimorar o desempenho dos classificadores, buscando melhores índices de acurácia e outras métricas de avaliação. Este trabalho também traçou uma comparação entre classificadores com dados **com** e **sem** tratamentos de sintaxe em seu corpus de treinamento. Assim, a etapa (ii) do pré-processamento só se aplica para modelos treinados **com** tratamentos lexicais, enquanto a etapa (i) se aplicou a todos os modelos (exceto as LLMs, que não precisam de qualquer processo de encriptação para entendimento da linguagem).

### 4.2.1 Tratamentos lexicais

O *dataset* natural foi concatenado com o *dataset* sintético, formando um *dataset* com 3878 amostras que foi utilizado no treinamento dos modelos. Antes do treinamento, algumas técnicas de pré-processamento foram empregadas no *dataset*, com o objetivo de melhorar a acurácia do modelo e também preparar os dados para atender os requisitos dos algoritmos.

No sentido de melhorar a acurácia dos modelos, foram empregadas técnicas de tratamento lexical nos comentários do *dataset*. Essas técnicas estão exibidas na Tabela 6. Lematização é outra técnica de tratamento lexical que pode ser observada na tabela. Contudo, observando empiricamente a acurácia dos modelos com e sem esse tratamento, optou-se por não utilizá-lo, embora, em alguns casos, ele possa aprimorar tanto a acurácia quanto o desempenho de modelos.

Tabela 6 – Tratamento lexicais aplicados no *dataset*

Tratamento lexical	Comentário com tratamento
Sem tratamento	Amo essa música! Estou encantado! kkkkkk :)
Subs. de expressão regular	Amo essa música! Estou encantado! risos :)
Substituição de emotes	Amo essa música! Estou encantado! kkkkkk felicidade
Concatenação de sinônimos	Amo essa música! Estou encantado feliz! kkkkk :)
Remoção de stopwords	Amo música! encantado! kkkkk :)
Lematização	Amar música! Estar encantar! kkkkk :)
Resultado final	Amo música! encantado feliz! risos felicidade

Fonte: elaborado pelo autor (2025).

Todas essas técnicas realizam mudanças no texto, aumentando a emoção aparente dos comentários. A técnica de substituição de expressões regulares atinge palavras com um certo padrão de repetição. Neste trabalho, o foco foram risadas como: kkkkk, hahahaha ou rrsrrs, e foram substituídas pela palavra *risos*. Esse procedimento foi realizado para que o modelo entendesse concatenações de qualquer tamanho das expressões regulares k, ha, rs, como risos, evitando que risadas de tamanhos distintos fossem entendidas como palavras distintas. As técnicas de Substituição de Emotes e Concatenação de sinônimos são usualmente utilizados para o sistema convergir mais rápido com uma menor quantidade de dados, reduzindo a ambiguidade em frases dupla, ou triplamente, diretas. Por exemplo, na frase "*Amo essa música! Estou encantado! kkkkkk :)*", a noção de felicidade é transmitida quatro vezes: "*Amo(1) essa música! Estou encantado(2)! kkkkkk(3) :)(4)*". Essa técnica vai priorizar o aprendizado para textos mais claros, colocando mais ambiguidade em textos menos explícitos. Um exemplo pode ser visto na Tabela 7.

Tabela 7 – Exemplos de ambiguidade ampliada após tratamentos lexicais

Tratamento lexical	Comentário	Classificação
Sem tratamento	Amo essa música! Estou encantado! kkkkkk :)	felicidade
Com tratamento	Amo felicidade essa música! Estou encantado felicidade! kkkkk felicidade	felicidade
Sem tratamento	:D Rindo para não chorar kkkk que bad!	tristeza
Com tratamento	felicidade Rindo felicidade para não chorar tristeza felicidade que bad tristeza	felicidade

Fonte: elaborado pelo autor (2025).

Além disso, optou-se pela técnica de Concatenação de sinônimos no lugar de substituição, acrescentando a palavra de emoção expressa relativa a palavra de sinônimo (por exemplo encantado, acrescenta-se felicidade), assim aumentando a generalidade do modelo, em detrimento de seu tempo de treinamento, embora, nesse e em muitos casos esse aumento não seja perceptível.

#### 4.2.2 TF-IDF

*Term Frequency - Inverse Document Frequency* (TF-IDF) é uma técnica de vetorização de texto que transforma dados em linguagem natural em representações numéricas compreensíveis por algoritmos de aprendizado de máquina. No presente trabalho, o TF-IDF foi utilizado para converter os textos das variáveis *Comentário*, *Emoção 1* e *Emoção 2* em vetores numéricos, viabilizando sua aplicação nos classificadores SVM e RNA.

A implementação do TF-IDF foi realizada por meio da biblioteca *scikit-learn* de Pedregosa *et al.* (2011), amplamente utilizada em aplicações de aprendizado supervisionado. Além disso, para o treinamento do modelo BERT, utilizou-se o *BERTTokenizer*, disponibilizado pela biblioteca *Transformers*, da plataforma *Hugging Face*, sendo este responsável por realizar a tokenização compatível com o modelo pré-treinado.

Nas abordagens baseadas em LLMs, não foi necessária a aplicação de técnicas explícitas de tokenização, uma vez que esses modelos são capazes de processar diretamente entradas em linguagem natural. Já para a avaliação humana, baseada na classificação de sentimentos avaliada por humanos via aplicação de formulários, o conjunto de dados foi utilizado em sua forma original, sem qualquer pré-processamento lexical ou tokenização.

### 4.3 Treinamento dos modelos

Dentre os classificadores testados neste trabalho, apenas as LLMs e a avaliação humana via preenchimento de formulários não receberam treinamento especializado. Descreveremos, portanto, o processo de treinamento dos modelos SVM, RNA e BERT, incluindo seus respectivos parâmetros e outros detalhes relevantes, como a divisão dos conjuntos de treino e teste para cada algoritmo. Os três classificadores foram treinados no *Google colab*, um serviço em nuvem do *Google* que permite a escrita e execução de código python em um ambiente de *notebook Jupyter*. O código dos modelos está disponível no **repositório**.

#### 4.3.1 SVM

O SVM é um dos modelos clássicos de IA utilizados em tarefas de classificação. Neste trabalho, foram treinados três classificadores SVM, utilizando diferentes bases de dados: (i) apenas o *dataset* natural, (ii) o *dataset* natural com pré processamento das amostras, e (iii) o *dataset* natural adicionado do *dataset* sintético, ambos com o pré-processamento de dados. Em todos os cenários, foi utilizado um kernel do tipo linear. O modelo foi implementado pela biblioteca *scikit-learn*. Outros hiperparâmetros estão apresentados na Tabela 8

Tabela 8 – Hiperparâmetros utilizados no SVM

Hiperparâmetro	Valor
C	1.0
Kernel	linear
Degree	10
Gamma	scale
Shrinking	True
Probability	False (unirrótulo)
Probability	True (multirrótulo)
random_state	42 (multirrótulo)

Fonte: elaborado pelo autor (2025).

Além disso, adotaram-se duas abordagens de classificação: unirrótulo e multirrótulo. Na abordagem multirrótulo, o modelo foi treinado para prever até duas emoções por comentário, utilizando como rótulos as colunas *Emoção 1* e *Emoção 2*. Para isso, a tarefa foi estruturada como uma classificação binária para cada possível emoção, permitindo que uma mesma entrada fosse associada a múltiplos rótulos simultaneamente.

### 4.3.2 *Multilayer Perceptron*

O segundo classificador empregado neste trabalho foi um modelo de RNA. Dentre as diversas arquiteturas existentes, optou-se pela utilização de uma Rede Neural do tipo Perceptron Multicamadas (MLP), aplicada às tarefas de classificação uni e multirrótulo de emoções. O modelo foi implementado utilizando a biblioteca *TensorFlow Keras* de Abadi *et al.* (2015), e seus hiperparâmetros foram definidos de forma empírica, com base em experimentação e validação nos dados. Os hiperparâmetros utilizados no modelo unirrótulo, que obteve maiores taxas de acurácia, encontram-se na Tabela 9.

Tabela 9 – Hiperparâmetros utilizados no MPL

Hiperparâmetro	Valor
Conexão entre camadas	dense
nº neurônios na 1ª camada	128
função de ativação (1ª camada)	gelu
Taxa de dropout	0.3
nº de neurônios na 2ª camada	64
função de ativação (2ª camada)	gelu
nº de neurônios na camada de saída	6
função de ativação (camada de saída)	softmax
Algoritmo de otimização	ADAM
Função de perda	sparse_categorical_crossentropy
nº de Épocas	5

Fonte: elaborado pelo autor (2025).

O modelo de RNA utilizado apresenta, ao todo, três camadas, sendo, portanto, caracterizado como um modelo multicamadas. A última camada, denominada camada de saída, possui número de neurônios proporcional à quantidade de rótulos considerados na classificação, seis. As conexões entre as camadas são do tipo totalmente conectadas (dense), ou seja, cada neurônio de uma camada está ligado a todos os neurônios da camada anterior. A função de ativação utilizada entre as camadas foi a GELU, escolhida por sua capacidade de suavizar transições e melhorar a performance em tarefas de processamento de linguagem natural.

### 4.3.3 *BERT*

O terceiro modelo de classificação aplicado neste trabalho foi o BERT, um modelo de processamento de linguagem natural desenvolvido pela *Google*, baseado na arquitetura Transformer. Esse tipo de modelo é caracterizado por sua leitura bidirecional, ou seja, a análise do texto ocorre tanto da esquerda para a direita quanto da direita para a esquerda, permitindo

uma compreensão mais profunda do contexto linguístico.

Para esta aplicação, utilizou-se o modelo pré-treinado *BERTimbau – Portuguese BERT*, proposto por Souza *et al.* (2019), treinado por 4 épocas. O conjunto de dados foi dividido em subconjuntos de treino e teste na proporção de 80% para treino e 20% para teste, utilizando a biblioteca *scikit-learn*.

#### 4.3.4 LLMs

Por fim, também foi utilizada uma LLM como classificador. Nessa etapa, o modelo ChatGPT 4 foi empregado para classificar 530 comentários, selecionados aleatoriamente a partir do *dataset* natural. Esta classificação foi realizada utilizando a abordagem Zero Shot, o que significa que o modelo não passou por um treinamento com exemplos específicos para esta tarefa. Adicionalmente, essa classificação seguiu os seis rótulos de emoção propostos por Ekman, acrescidos de uma categoria para emoção neutra. Os comentários foram avaliados 10 a 10 pela LLM, utilizando o Prompt: "*Avalie as 10 frases abaixo. Cada frase deve conter dois rótulos de uma das sete categorias: felicidade, raiva, tristeza, medo, nojo, surpresa e neutro. O segundo rótulo pode estar vazio, denotado por '-'*".

#### 4.3.5 Avaliação Humana

Visando complementar os classificadores desenvolvidos com inteligência artificial, foi conduzida uma pesquisa com 25 participantes. O objetivo principal desta etapa foi estabelecer um "classificador humano médio" para servir como referência.

Para tanto, os participantes foram instruídos a rotular de 21 amostras de comentários, extraídas aleatoriamente do *dataset* natural. A classificação dessas sentenças considerou sete categorias de emoções, baseadas no modelo de emoções básicas proposto por Paul Ekman, mais a emoção neutra. Os resultados da pesquisa foram comparados com os rótulos originais das 21 amostras. Os resultados da pesquisa se encontram na íntegra no **repositório**.

### 4.4 Considerações finais

Neste capítulo, foi apresentada a concepção do corpus natural, composto por comentários extraídos de redes sociais e rotulados manualmente. Também foi evidenciada a necessidade da construção de um corpus sintético para complementar as classes minoritárias do corpus na-

tural. Em seguida, foram descritos os processos de pré-processamento que, em alguns casos, foram aplicados aos dados de ambos os *datasets*, assim como seus impactos no desempenho dos modelos. Por fim, foram detalhados os classificadores utilizados (SVM, MLP, BERT e LLM), com destaque para os hiperparâmetros adotados, os valores de sobreajuste observados, e o processo de estimativa de desempenho de um classificador humano por meio de aplicação de formulários. No próximo capítulo, apresentamos os resultados de acurácia dos classificadores, assim como outras métricas relevantes para o entendimento de sua qualidade.

## 5 RESULTADOS

A classificação de emoções neste trabalho utilizou cinco modelos e duas abordagens distintas. No universo dos modelos, foram empregados o SVM (um modelo clássico), o modelo perceptron multicamadas (RNA), o modelo BERT, uma LLM por meio do ChatGPT e a classificação humana, obtida através de respostas em um formulário. As abordagens de classificação de emoções dividiram-se em duas: unirrótulo e multirrótulo. Na abordagem unirrótulo, cada frase foi associada a uma única emoção do conjunto de emoções discretas de Ekman. Por outro lado, na abordagem multirrótulo, as frases puderam receber um ou dois rótulos. Três abordagens para o dataset foram utilizadas para o treinamento dos modelos: (i) Dataset natural pré-processado (tratamentos lexicais) mais o dataset sintético; (ii) Dataset natural pré-processado; (iii) Dataset natural sem pré-processamento dos dados.

### 5.1 Resultados da classificação unirrótulo

Nesta subseção vamos comentar os resultados dos modelos unirrótulo do SVM, Perceptron multicamadas, BERT, LLM e pesquisa humana.

#### 5.1.1 Avaliação do SVM

A avaliação do modelo SVM pode ser observada no Quadro 1. Podemos notar um aumento pequeno de acurácia em relação ao modelo com e sem a assistência de dados sintéticos. Um aumento maior pode ser observado do modelo após ser treinado com os dados pré-processados. O maior ganho obtido pelo treinamento com a adição de dados sintéticos se dá pelo acerto em classes minorizadas, como podemos perceber pelo aumento significativo da taxa de Recall da linha 2 para a linha 1 do quadro.

Quadro 1 – Métricas do modelo de SVM

Conjunto de treino	Acurácia	F1-score	Recall
Dataset natural pré-processado + sintético	56,81%	51,36%	49,66%
Dataset natural pré-processado	55,50%	45,16%	40,83%
Dataset natural sem pré-processamento	52,21%	44,33%	39,33%

Fonte: elaborado pelo autor.

### 5.1.2 Avaliação do modelo Perceptron Multicamadas

As métricas atribuídas ao modelo Perceptron Multicamadas estão apresentadas no Quadro 2. Embora esse modelo tenha registrado acurácias inferiores em comparação aos demais, as técnicas de pré-processamento e a inclusão de dados sintéticos contribuíram para um aumento em seu desempenho.

Quadro 2 – Métricas do modelo Perceptron multicamadas

Conjunto de treino	Acurácia	F1-score	Recall
Dataset natural pré-processado + sintético	53,69%	50,00%	50,00%
Dataset natural pré-processado	52,55%	41,00%	39,00%
Dataset natural sem pré-processamento	48,77%	37,00%	36,14%

Fonte: elaborado pelo autor.

### 5.1.3 Avaliação do modelo BERT

As métricas atribuídas ao modelo BERT estão apresentadas no Quadro 3. O ganho com o acréscimo do dataset sintético não se reflete somente em um aumento perceptível da acurácia, mas também da generalização do modelo, como podemos perceber pelo incremento das métricas F1-score e Recall.

Quadro 3 – Métricas do modelo BERT

Conjunto de treino	Acurácia	F1-score	Recall
Dataset natural sem pré-processamento + sintético	73,09%	74,50%	73,00%
Dataset natural sem pré-processamento	65,55%	54,00%	52,00%

Fonte: elaborado pelo autor.

### 5.1.4 Avaliação da LLM

O Quadro 4 apresenta as métricas de avaliação empregadas para avaliar a classificação de emoções por meio de uma LLM (ChatGPT). Para essa classificação foram utilizados 530 exemplos do dataset natural. Não foram testadas versões do dataset com pré-processamento de texto, uma vez que LLMs estão preparadas para lidar com processamento linguagem natural. Também não foi utilizado o dataset expandido com dados sintéticos gerados pela própria LLM, a fim de evitar uma classificação com maiores vieses.

Quadro 4 – Métricas da LLM

Conjunto de treino	Acurácia	F1-score	Recall
Dataset natural sem pré-processamento	71,88%	66,22%	74,26%

Fonte: elaborado pelo autor.

### 5.1.5 Avaliação humana

As métricas de avaliação da classificação de sentimentos realizada por participantes humanos, por meio do preenchimento de formulários, está apresentada no Quadro 5.

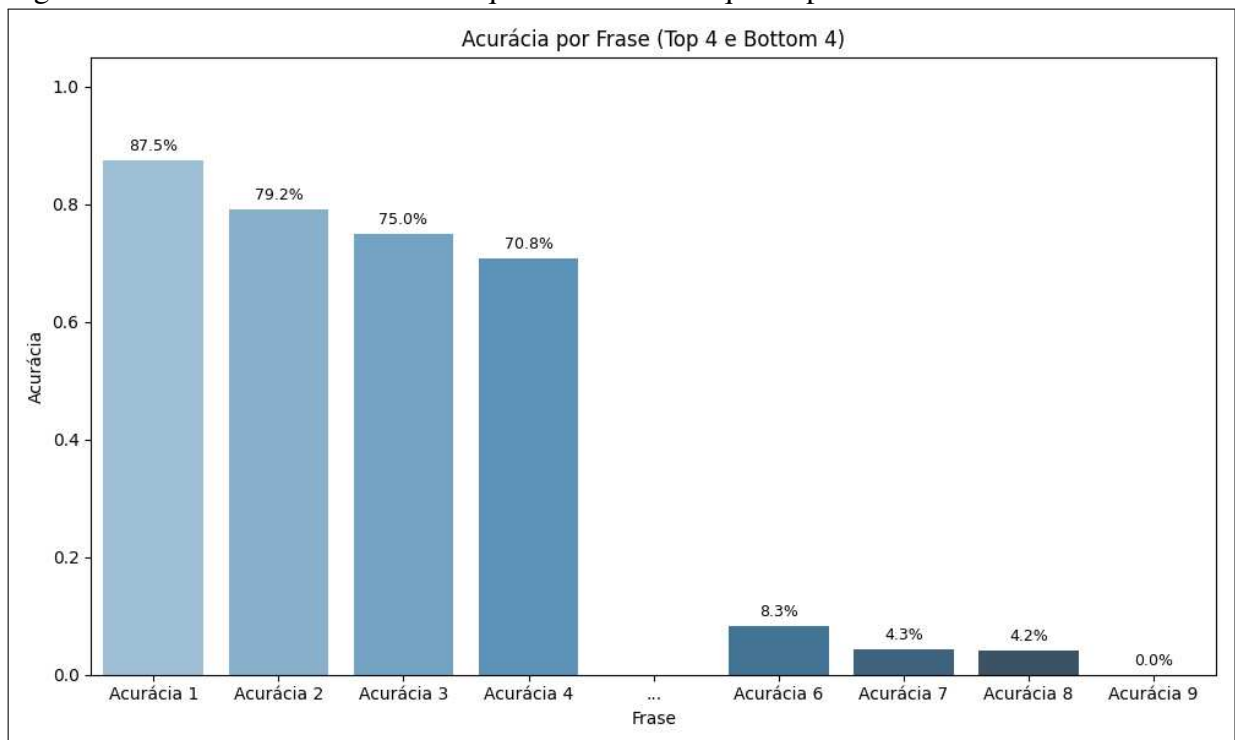
Quadro 5 – Métricas das avaliações via Formulário

Conjunto de treino	Acurácia	F1-score	Recall
Dataset natural sem pré-processamento	43,65%	39,18%	42,47%

Fonte: elaborado pelo autor.

Essa classificação apresentou os piores resultados entre os cinco modelos analisados. No entanto, observando alguns indicadores, é possível notar a presença de *outliers*. A Figura 6 apresenta os resultados de acurácia das quatro frases com maior índice de acerto e das quatro com menor desempenho. Nota-se uma variabilidade significativa entre essas categorias, evidenciando uma discrepância nas taxas de acerto por frase.

Figura 6 – Divisão das acurácias das quatro melhores e quatro piores frases avaliadas



Fonte: elaborado pelo autor (2025).

A variância da acurácia entre as 21 frases foi calculada utilizando a equação  $\sigma^2 =$

$\frac{\sum_{i=1}^N (x_i - \mu)^2}{N}$ , resultando em um valor de 0,0721. Esse número reforça a presença de oscilações relevantes no desempenho da classificação humana, indicando que alguns comentários foram interpretados de forma muito mais consistente do que outros.

Foram calculados métricas de acurácia, considerando os resultados das quatro piores frases como *outliers* e ignorando-os, obtendo o resultado de 53,21%.

## 5.2 Resultados da classificação multirrótulo

Nesta subseção, discutimos os resultados dos modelos multirrótulo: SVM, Perceptron Multicamadas e LLM. Para esses modelos, foram adotadas duas métricas de avaliação mais adequadas ao cenário multirrótulo, diferentes das utilizadas na abordagem unirrótulo: a Índice de Jaccard e o *Hamming Loss*. Essas métricas são mais apropriadas para medir o desempenho de classificadores que podem atribuir múltiplos rótulos simultaneamente a uma mesma instância.

### 5.2.1 Avaliação do modelo SVM multirrótulo

As métricas de avaliação do modelo SVM multirrótulo estão destacados no Quadro 6. Com os tratamentos lexicais e o uso de dados sintéticos, percebemos melhoras nas quatro métricas, com quedas constantes do Hamming Loss e crescimento acentuado da Índice de Jaccard e do F1-Score.

Quadro 6 – Métricas do modelo SVM Multirrótulo

Conjunto de treino	Índice de Jaccard	F1-score	Recall	Hamming Loss
Dataset natural pré-processado + sintético	50,66%	61,99%	50,81%	09,88%
Dataset natural pré-processado	47,13%	53,24%	43,56%	12,20%
Dataset natural sem pré-processamento	39,74%	48,41%	40,32%	13,54%

Fonte: elaborado pelo autor.

### 5.2.2 Avaliação do modelo Perceptron Multicamadas multirrótulo

As métricas de avaliação do modelo Perceptron multicamadas multirrótulo estão destacados no Quadro 7.

### 5.2.3 Avaliação da LLM utilizando as métricas de multirrótulo

Além do treinamento dos modelos multirrótulo, também foi avaliado o desempenho de uma LLM (ChatGPT) na tarefa de classificação de frases do dataset natural sob a abordagem

Quadro 7 – Métricas do modelo Perceptron multicamadas Multirrótulo

Conjunto de treino	Índice de Jaccard	F1-score	Recall	Hamming Loss
Dataset natural pré-processado + sintético	55,10%	63,10%	55,17%	10,32%
Dataset natural pré-processado	47,13%	53,24%	43,56%	12,20%
Dataset natural sem pré-processamento	42,36%	47,69%	42,63%	14,07%

Fonte: elaborado pelo autor.

multirrótulo. O Quadro 8 apresenta as métricas de desempenho da LLM nesse contexto.

Quadro 8 – Métricas da LLM Multirrótulo

Conjunto de treino	Índice de Jaccard	F1-score	Recall	Hamming Loss
Dataset natural	69,91%	71,64%	69,91%	08,81%

Fonte: elaborado pelo autor.

### 5.3 Análise dos modelos unirrótulo

Um dos objetivos deste trabalho foi investigar como diferentes modelos de inteligência artificial, pertencentes a distintos paradigmas, se comportam na tarefa de classificação de emoções. Foram utilizados três tipos de classificadores: um modelo de aprendizado de máquina tradicional (SVM), um modelo de rede neural artificial (Perceptron Multicamadas) e um modelo baseado em linguagem natural profunda (BERT), mais a avaliação de sentimentos da LLM. Para todos os classificadores unirrótulo treinados, o melhor desempenho foi obtido com a combinação do dataset natural e do dataset sintético, ambos submetidos a técnicas de pré-processamento.

O Quadro 9 apresenta os índices de acurácia obtidos por cada modelo nesse cenário.

Quadro 9 – Acurácia dos modelos utilizados nesse trabalho

Modelo	Acurácia
SVM	56,81%
Perceptron Multicamadas	53,69%
BERT	73,09%
LLM	74,26%

Fonte: elaborado pelo autor.

As métricas de acurácia revelaram-se promissoras, evidenciando que modelos clássicos como o SVM e o Perceptron multicamadas mesmo ignorando conseguem extrair características do texto que determinem seus rótulos em aproximadamente 50% dos casos. Enquanto o BERT, utilizando contexto, conseguiu resultados satisfatórios. Além disso, todos os modelos treinados superaram a linha de base inicial, estabelecida pela avaliação humana por meio de formulários (53,21 % após a remoção de *outliers*). O SVM alcançou 56,81 % de acerto, enquanto a RNA obteve 53,69 %. O modelo baseado em BERT apresentou o melhor desempenho geral, se

aproximando inclusive da LLM avaliada em um cenário com poucas amostras.

#### 5.4 Análise dos modelos multirrótulo

Além da abordagem unirrótulo, este trabalho também investigou o comportamento de diferentes modelos de Inteligência Artificial na tarefa de classificação de emoções sob a perspectiva multirrótulo. Para essa finalidade, foram treinados dois modelos: SVM e Perceptron Multicamadas, ambos adaptados à abordagem multirrótulo. Adicionalmente, avaliou-se o desempenho de modelos baseados em LLM nesse tipo de tarefa. O Quadro 10 apresenta os resultados de índice de Jaccard e *Hamming Loss* obtidos pelos três classificadores.

Quadro 10 – Índice de Jaccard e *Hamming Loss* dos modelos multirrótulo utilizados nesse trabalho

Modelo	Índice de Jaccard	<i>Hamming Loss</i>
SVM	50,66%	09,88%
Perceptron Multicamadas	55,10%	10,32%
LLM	69,91%	08,81%

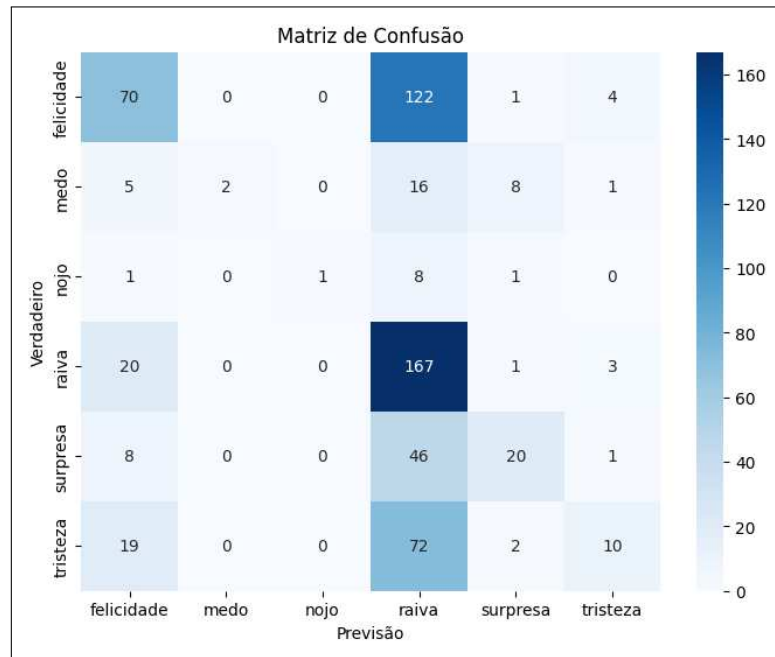
Fonte: elaborado pelo autor.

Ambos os modelos treinados apresentaram evoluções aos tratamentos lexicais aplicados e a inclusão de corpus sintético. No entanto, quando comparados ao desempenho da LLM, observa-se uma discrepância significativa na qualidade da classificação. Ainda assim, esses modelos se mostram como alternativas viáveis em cenários com restrições computacionais, menor disponibilidade de dados ou em tarefas menos críticas, nas quais um nível de acurácia não refinado seja aceitável.

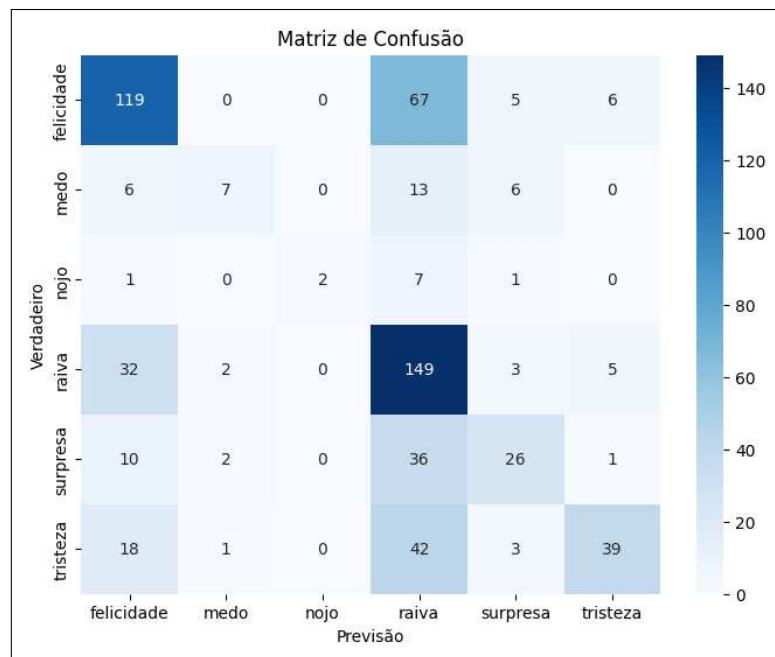
#### 5.5 Tratamentos Lexicais dos comentários

Nesta sessão vamos analisar como os tratamentos lexicais empregados nas amostras cumularam em índices de acurácia superiores. As Figuras 7 e 8, representam a matriz de confusão do modelo SVM após os treinamento **sem** tratamento de dados e **com**, respectivamente.

A análise das figuras permite observar um aumento no índice de acertos do modelo treinado com um corpus submetido a tratamentos lexicais. Nota-se, em especial, uma redução no viés do modelo, que anteriormente apresentava tendência à classificação predominante na classe majoritária (raiva). Após os tratamentos, os resultados passaram a se concentrar principalmente nas classes **raiva** e **felicidade**. A quantidade de verdadeiros positivos para a classe felicidade apresentou um crescimento de 58%. Esse efeito revela um aumento do viés individual por

Figura 7 – Matriz de confusão do modelo sem tratamento do *dataset*

Fonte: elaborado pelo autor (2025).

Figura 8 – Matriz de confusão do modelo com tratamento do *dataset*

Fonte: elaborado pelo autor (2025).

classe: ou seja, um comentário lexicalmente mais próximo de determinada classe tende a ser classificado como pertencente a ela. Embora essa tendência contribua para o aumento da média geral de acertos, ela também introduz um efeito colateral importante, a classificação deliberadamente ignora o contexto em alguns casos, levando, por exemplo, à categorização incorreta de comentários irônicos ou ambíguos. As matrizes de confusão do SVM e do BERT estão disponíveis no Apêndice A. Em ambas as matrizes é possível notar o aumento de acurácia

e viés de classe.

Também podemos fazer essa análise por meio das métricas, como F1-score e Recall. O Quadro 11 apresenta as métricas para os modelos.

## 5.6 Dados sintéticos

Também foi investigado o uso de dados sintéticos para melhorar a generalização dos modelos. No Capítulo 4, foram apresentadas as especificações do corpus natural, evidenciando a existência de classes majoritárias. O *dataset* sintético foi criado para complementar o corpus natural, adicionando amostras das classes minoritárias. As Figuras 9 e 10 mostram, respectivamente, as matrizes de confusão da classificação do modelo SVM sem e com a inclusão dos dados sintéticos.

Observa-se uma maior generalização do modelo após a inclusão do *dataset* sintético, refletida pela distribuição mais equilibrada dos elementos na matriz de confusão. Nota-se um aumento no número de elementos na diagonal principal (classificações corretas), assim como um crescimento nos elementos fora da diagonal (erros de classificação). Esse comportamento indica uma maior capacidade de generalização do modelo e uma redução do viés, evidenciada pela diminuição das classificações concentradas nas classes majoritárias.

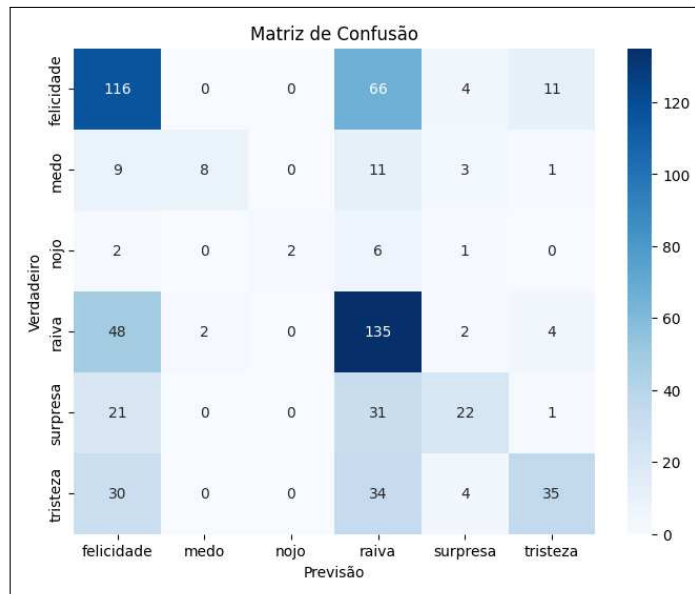
Também podemos notar generalização do modelo após o treinamento com dados sintéticos observando as métricas como F1-score e Recall no Quadro 11. As Figuras 11 e 12 apresentam as métricas de F1-score e Recall, respectivamente.

Quadro 11 – Métricas de generalização dos modelos

Modelo	F1-score	Recall
SVM sem dados sintéticos	45,16%	40,83%
SVM com dados sintéticos	51,36%	49,66%
Perceptron Multicamadas sem dados sintéticos	41,00%	39,00%
Perceptron Multicamadas com dados sintéticos	50,00%	50,00%
BERT sem dados sintéticos	54,00%	52,00%
BERT com dados sintéticos	74,50%	73,00%

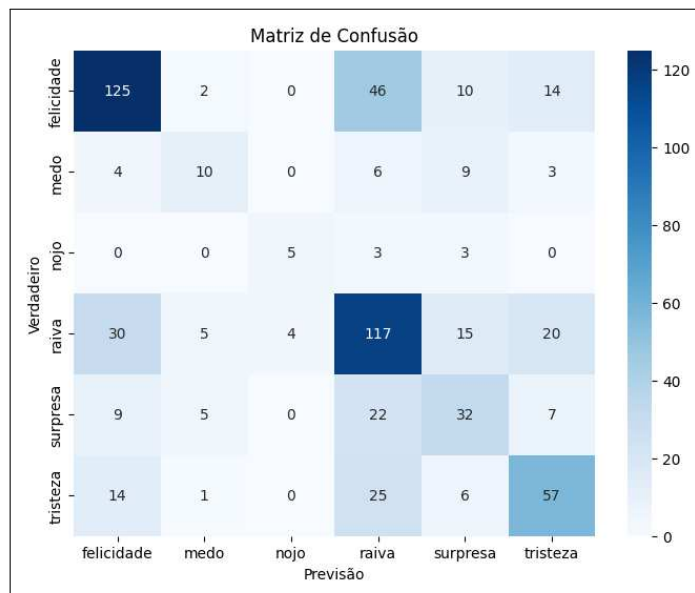
Fonte: elaborado pelo autor.

Figura 9 – Matriz de confusão do SVM com tratamento do *dataset*



Fonte: elaborado pelo autor (2025).

Figura 10 – Matriz de confusão do SVM com tratamento do *dataset* e *dataset* sintético



Fonte: elaborado pelo autor (2025).

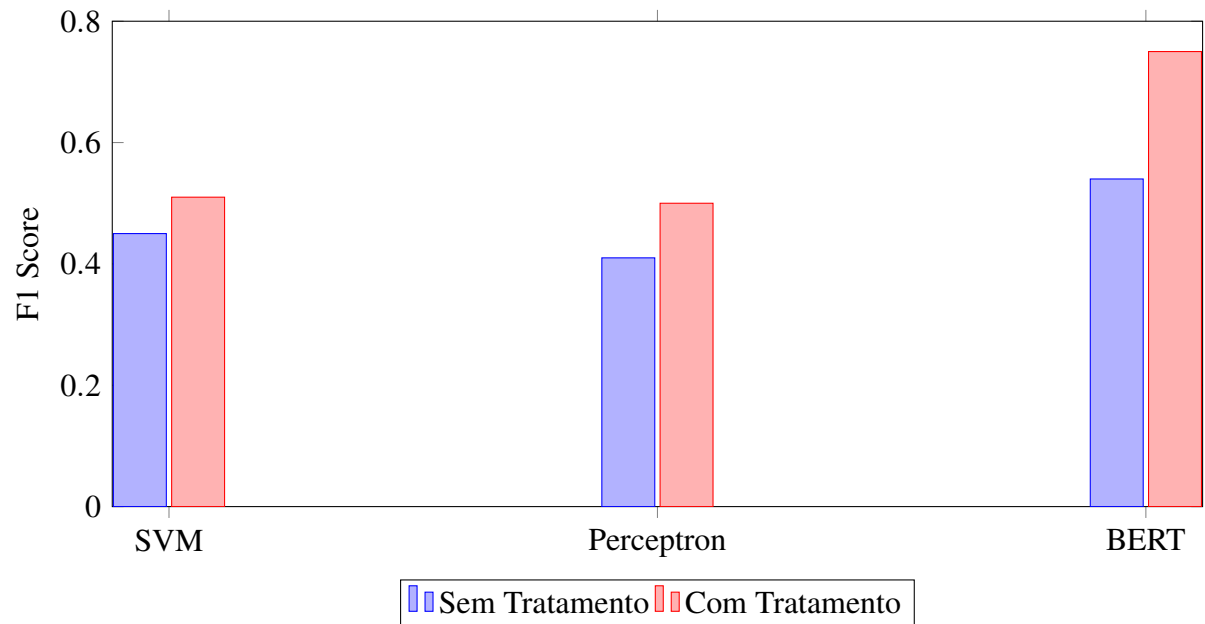


Figura 11 – Métricas de F1-Score para os algoritmos SVM, Perceptron e BERT

Elaborado pelo autor (2025).

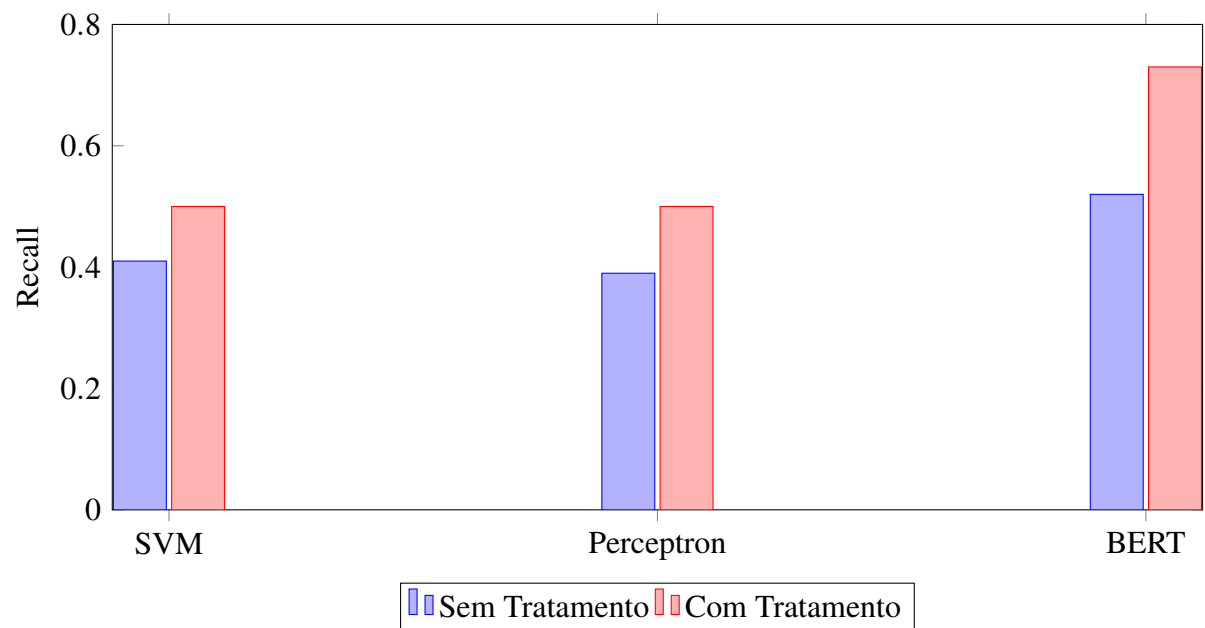


Figura 12 – Métricas de Recall para os algoritmos SVM, Perceptron e BERT

Elaborado pelo autor (2025).

## 6 CONCLUSÕES E TRABALHOS FUTUROS

Este trabalho investigou a classificação de emoções em comentários curtos utilizando diferentes modelos, paradigmas e técnicas. Foram avaliados quatro classificadores distintos: SVM, Perceptron Multicamadas, BERT, um LLM representado pelo ChatGPT, e a avaliação humana por meio de formulários. Em alguns casos, a tarefa de classificação variou entre os métodos unirrótulo e multirrótulo. Além disso, dados sintéticos foram integrados e analisados com o objetivo de aprimorar a qualidade dos classificadores.

A tarefa de classificar emoções extraídas de textos curtos revelou-se desafiadora e repleta de características que exigem uma análise cuidadosa. Um exemplo disso está na avaliação humana das frases por meio de formulários, que apresentou uma grande quantidade de rotulagens divergentes tanto entre si quanto em relação ao rótulo proposto neste estudo. Esse fenômeno resultou em baixas taxas de acurácia para a tarefa. Modelos clássicos, como o SVM e o Perceptron multicamadas, também enfrentaram dificuldades, alcançando uma acurácia de aproximadamente 50%, ou seja, acertando apenas metade das amostras, tanto na abordagem de classificação unirrótulo quanto multirrótulo. Ainda assim, nota-se que esses modelos conseguem extrair características dos textos que lhes permitem classificar corretamente parte das sentenças, mesmo desconsiderando aspectos relevantes como o contexto e a interconexão entre as palavras. Por outro lado, o modelo pré-treinado BERT obteve métricas mais satisfatórias, comparáveis às de grandes modelos, como a LLM *ChatGPT* enquanto esta realiza uma classificação na abordagem *zero-shot*.

### 6.1 Limitações deste trabalho e trabalhos futuros

Nesta seção, apresentamos os pontos fracos e as dificuldades encontradas, além de exploraremos as possibilidades de desenvolvimento deste trabalho, detalhando como os problemas identificados podem ser contornados.

#### 6.1.1 Limitações

O *dataset* natural utilizado conta com 2.209 amostras recolhidas de diversas plataformas de redes sociais. Esse *dataset* foi construído sob medida para os experimentos propostos, com amostras rotuladas manualmente com diversos atributos, sendo voltado principalmente ao estudo da classificação de emoções com base na teoria das emoções discretas de Ekman. Todavia,

como verificado no Capítulo 5 (Resultados), o número de amostras mostrou-se enviesado em relação a algumas classes marjoritárias e limitado em termos de diversidade, o que reduziu a capacidade de generalização dos modelos. Outros trabalhos exploraram *datasets* de emoções mais amplos, por exemplo, (KANSAON *et al.*, 2019) utilizou um *dataset* com 175.116 tweets para a classificação de emoções.

Além do *dataset* natural, o *dataset* sintético também apresentou uma quantidade limitada de amostras: foram 1.669 exemplos gerados com o auxílio de diversas LLMs gratuitas. A análise de qualidade comparando amostras geradas por LLMs gratuitas e pagas não foi realizada neste estudo.

A adoção da teoria das emoções discretas de Ekman como base para os rótulos dos modelos de classificação também impõe algumas limitações. Apesar de amplamente utilizada em trabalhos relacionados à classificação de emoções, a teoria de Ekman fundamenta-se em expressões faciais universais, enquanto este trabalho concentra-se na análise de comentários textuais em redes sociais. As emoções de Ekman podem ser agrupadas em três categorias: positivas (felicidade), negativas (raiva, tristeza, medo e nojo) e ambíguas (surpresa). O número reduzido de emoções positivas e ambíguas nessa teoria contribui para o desbalanceamento de classes nas *features* Sentidos e Emoções 1/Emoções 2.

No Capítulo 5 (Resultados) foram apresentadas as métricas de avaliação dos modelos de classificação. É importante observar, contudo, que o conjunto de amostras avaliadas pelas LLMs, para cálculo de acurácia e demais métricas, e o submetido à validação humana foi menor que o utilizado na avaliação dos outros modelos. Nessas duas validações foram usadas 500 amostras, enquanto os demais modelos foram avaliados com conjuntos que variaram de 507 a 925 amostras.

### 6.1.2 *Trabalhos Futuros*

Em trabalhos futuros, os corpora natural e sintético de emoções podem ser expandidos. Neste trabalho, os comentários oriundos de redes sociais foram coletados e catalogados manualmente, enquanto os comentários gerados por LLM utilizaram a abordagem **zero-shot**, ou seja, sem qualquer treinamento ou adaptação prévia dos modelos à tarefa proposta.

Para aprimorar a obtenção de dados naturais, recomenda-se a exploração de *Application Programming Interface* (APIs), interfaces de disponibilização de dados, oferecidas por diversas plataformas. Muitas redes sociais disponibilizam APIs gratuitas ou por meio de planos

voltados a desenvolvedores, o que pode facilitar a coleta automatizada e em maior escala.

No que diz respeito à geração de dados sintéticos, técnicas de aprendizado prévio, como **n-shot** ou **few-shot learning**, podem ser exploradas como estratégias para melhorar a qualidade e a adequação dos exemplos gerados pelas LLMs, contribuindo assim para o aumento da precisão dos modelos de classificação.

A exploração de novos modelos teóricos de emoções é uma possibilidade relevante para mitigar o problema de desbalanceamento das classes nas *features* Sentido e Emoção 1/Emoção 2, um reflexo direto das limitações da teoria das emoções básicas de Ekman. Em sua revisão sistemática, (KHARE *et al.*, 2024) aborda as principais teorias de emoções amplamente utilizadas na literatura científica.

Um modelo alternativo à abordagem de Ekman é a **Roda de Plutchik** (PLUTCHIK, 1980), que, embora também seja uma teoria discreta das emoções, oferece uma estrutura mais distribuída e rica. A Roda propõe oito emoções básicas (**medo, tristeza, confiança, raiva, surpresa, nojo, felicidade e antecipação**), organizadas em um círculo, no qual emoções compostas derivam da combinação ou intensidade das emoções básicas, sendo posicionadas em coordenadas específicas desse espaço.

Além das teorias discretas, existem abordagens multidimensionais das emoções, como a **Teoria das Emoções Multidimensionais**, que considera as emoções fenômenos mais complexos, melhor representados por dimensões contínuas. Um exemplo dessa abordagem é o **Espaço Emocional Bidimensional de Russell**, que utiliza dois eixos principais, a valência (agradável/desagradável) e ativação (alta/baixa energia), para mapear uma ampla variedade de estados emocionais.

## 6.2 Considerações finais e contribuições

Este trabalho apresentou uma análise exploratória do uso de diferentes modelos matemáticos e de linguagem para a tarefa de classificação de sentimentos em textos curtos. Foram investigadas duas abordagens distintas: unirrótulo e multirrótulo.

Além da avaliação dos modelos, este estudo também contribuiu com a concepção e disponibilização de dois corpus rotulados com sentimentos: um natural e outro sintético. As amostras do corpus sintético foram majoritariamente construídas para complementar o corpus natural, atuando especialmente no balanceamento das classes minoritárias.

Os resultados obtidos evidenciam os desafios de se avaliar sentimentos por meio de

textos encontrados em ambientes online, especialmente devido à subjetividade e à ambiguidade inerentes a esse tipo de conteúdo. No entanto, também demonstram que certos padrões, características e vieses podem ser identificados e utilizados pelos modelos para inferir, com razoável precisão, o sentimento aparente de um texto.

## REFERÊNCIAS

- ABADI, M.; AGARWAL, A.; BARHAM, P. *et al.* **TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems**. 2015. <<https://www.tensorflow.org/>>.
- ALM, C. O.; ROTH, D.; SPROAT, R. Emotions from text: Machine learning for text-based emotion prediction. In: **Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing (HLT/EMNLP)**. Vancouver, British Columbia, Canada: Association for Computational Linguistics, 2005. p. 579–586.
- ARISTÓTELES. **Retórica**. São Paulo: Editora Edipro, 2017. Obra original do século IV a.C.
- ASHRAF, N.; ASHRAF, N.; KHAN, L.; KHAN, L.; BUTT, S.; BUTT, S.; CHANG, H.-T.; CHANG, H.-T.; TSAI, T.-H.; SIDOROV, G.; SIDOROV, G.; GELBUKH, A.; GELBUKH, A. Multi-label emotion classification of urdu tweets. **PeerJ**, 2022.
- BHAT, S. **Emotion Classification in Short English Texts using Deep Learning Techniques**. 2024. Disponível em: <<https://arxiv.org/abs/2402.16034>>.
- BOSER, B. E.; GUYON, I. M.; VAPNIK, V. N. A training algorithm for optimal margin classifiers. In: **Proceedings of the Fifth Annual Workshop on Computational Learning Theory**. [S.l.]: ACM, 1992. p. 144–152.
- CHAWLA, N. V.; BOWYER, K. W.; HALL, L. O.; KEGELMEYER, W. P. Smote: synthetic minority over-sampling technique. **Journal of artificial intelligence research**, v. 16, p. 321–357, 2002.
- COVER, T. M. Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. **IEEE Transactions on Electronic Computers**, IEEE, EC-14, n. 3, p. 326–334, 1965.
- DAMASIO, A. R. **O erro de Descartes: emoção, razão e o cérebro humano**. São Paulo: Companhia das Letras, 1994.
- DEVLIN, J.; CHANG, M.-W.; LEE, K.; TOUTANOVA, K. Bert: Pre-training of deep bidirectional transformers for language understanding. **arXiv preprint arXiv:1810.04805**, 2018. Disponível em: <<https://arxiv.org/abs/1810.04805>>.
- D’SA, A. G.; ILLINA, I.; FOHR, D.; AKBAR, A. Exploration of multi-corpus learning for hate speech classification in low resource scenarios. In: SOJKA, P.; HORÁK, A.; KOPEČEK, I.; PALA, K. (Ed.). **Text, Speech, and Dialogue**. Cham: Springer International Publishing, 2022. p. 238–250.
- EKMAN, P. Universals and cultural differences in facial expressions of emotion. In: COLE, J. (Ed.). **Nebraska Symposium on Motivation**. Lincoln: University of Nebraska Press, 1972. v. 19, p. 207–283.
- FACELI, K.; LORENA, A.; GAMA, J.; ALMEIDA, T.; CARVALHO, A. **Inteligência Artificial Uma Abordagem de Aprendizado de Máquina**. [S.l.]: Grupo Gen, 2024. v. 2.
- FILIPE, S.; BATISTA, F.; RIBEIRO, R. Different Lexicon-Based Approaches to Emotion Identification in Portuguese Tweets. In: oES, A. S.; HENRIQUES, P. R.; QUEIRÓS, R. (Ed.). **9th Symposium on Languages, Applications and Technologies (SLATE 2020)**. Dagstuhl,

Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2020. (Open Access Series in Informatics (OASICs), v. 83), p. 12:1–12:8. ISBN 978-3-95977-165-8. ISSN 2190-6807. Disponível em: <<https://drops.dagstuhl.de/entities/document/10.4230/OASICs.SLATE.2020.12>>.

FORTUNA, P.; NUNES, S. A survey on automatic detection of hate speech in text. **ACM**, Association for Computing Machinery, New York, NY, USA, v. 51, n. 4, jul. 2018. ISSN 0360-0300. Disponível em: <<https://doi.org/10.1145/3232676>>.

FRIJDA, N. H. **The Emotions**. Cambridge: Cambridge University Press, 1986.

HAMMES, L.; FREITAS, L. Utilizando bertimbau para a classificação de emoções em português. In: **Anais do XIII Simpósio Brasileiro de Tecnologia da Informação e da Linguagem Humana**. Porto Alegre, RS, Brasil: SBC, 2021. p. 56–63. ISSN 0000-0000. Disponível em: <<https://sol.sbc.org.br/index.php/stil/article/view/17784>>.

KANG, X.; SHI, X.; WU, Y.; REN, F. Active learning with complementary sampling for instructing class-biased multi-label text emotion classification. **IEEE Transactions on Affective Computing**, v. 14, n. 1, p. 523–536, 2023.

KANSAON, D.; BRANDÃO, M.; PINTO, S. d. P. Análise de algoritmos de classificação para detecção de emoções em tweets em português brasileiro. **iSys - Brazilian Journal of Information Systems**, 2019.

KHARE, S. K.; BLANES-VIDAL, V.; NADIMI, E. S.; ACHARYA, U. Emotion recognition and artificial intelligence: A systematic review (2014-2023) and research recommendations. **Information Fusion**, v. 102, p. 102019, 2024.

LIANG, S. Comparative analysis of svm, xgboost and neural network on hate speech classification. **Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)**, v. 5, n. 5, p. 896 – 903, Oct. 2021. Disponível em: <<https://jurnal.iaii.or.id/index.php/RESTI/article/view/3506>>.

LIU, X.; SHI, T.; ZHOU, G.; LIU, M.; YIN, Z.; YIN, L.; ZHENG, W. Emotion classification for short texts: an improved multi-label method. **Humanities and Social Sciences Communications**, v. 2023, p. s41599–023, 06 2023.

PASUPA, K.; Seneewong Na Ayutthaya, T. Thai sentiment analysis with deep learning techniques: A comparative study based on word embedding, pos-tag, and sentic features. **Sustainable Cities and Society**, v. 50, p. 101615, 2019. ISSN 2210-6707. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S2210670718325228>>.

PAVLYSHENKO, B.; STASIUK, M. Using large language models for data augmentation in text classification models. **International Journal of Computing**, p. 148–154, 03 2025.

PEDREGOSA, F.; VAROQUAUX, G.; GRAMFORT, A. *et al.* Scikit-learn: Machine learning in python. **Journal of Machine Learning Research**, v. 12, p. 2825–2830, 2011.

PLUTCHIK, R. **Emotion: A Psychoevolutionary Synthesis**. New York: Harper Row, 1980.

QIN, Y.; SHI, Y.; HAO, X.; LIU, J. Microblog text emotion classification algorithm based on tcn-bigru and dual attention. **Information**, v. 14, p. 90, 02 2023.

ROSENBLAT, F. **DTIC AD0256582: PRINCIPLES OF NEURODYNAMICS. PERCEPTRONS AND THE THEORY OF BRAIN MECHANISMS.** [S.l.]: Defense Technical Information Center., 1961.

ROSENBLATT, F. The perceptron: a probabilistic model for information storage and organization in the brain. **Psychological Review**, American Psychological Association, v. 65, n. 6, p. 386–408, 1958.

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

SILVA, F.; FREITAS, L. Brazilian portuguese hate speech classification using bertimbau. **The International FLAIRS Conference Proceedings**, v. 35, May 2022. Disponível em: <<https://journals.flvc.org/FLAIRS/article/view/130594>>.

SILVA, L. d.; SANTOS, M. d. C.; COSTA, L. d. R.; PINTO, A. L. G. F.; BOSA, C. A. Empatia afetiva e cognitiva no transtorno do espectro autista (tea): uma revisão integrativa da literatura. **Revista Brasileira de Educação Especial**, SciELO Brasil, v. 27, n. 1, p. 155–172, 2021.

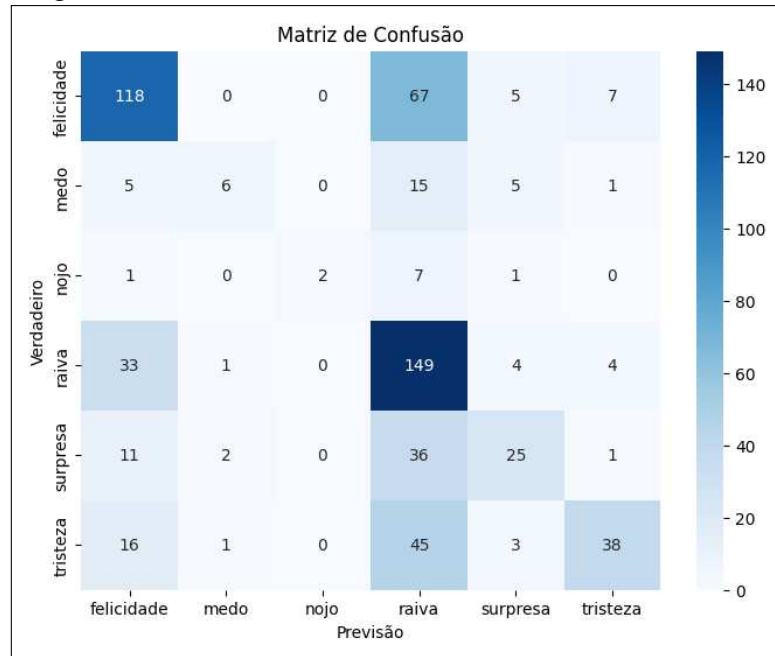
SOUZA, F.; NOGUEIRA, R.; LOTUFO, R. Portuguese named entity recognition using bert-crf. **arXiv preprint arXiv:1909.10649**, 2019. Disponível em: <<http://arxiv.org/abs/1909.10649>>.

VAPNIK, V. N. **The Nature of Statistical Learning Theory**. New York: Springer, 1995.

VASWANI, A.; SHAZEER, N.; PARMAR, N.; USZKOREIT, J.; JONES, L.; GOMEZ, A. N.; KAISER, L.; POLOSUKHIN, I. Attention is all you need. **arXiv preprint arXiv:1706.03762**, 2017. Disponível em: <<https://arxiv.org/abs/1706.03762>>.

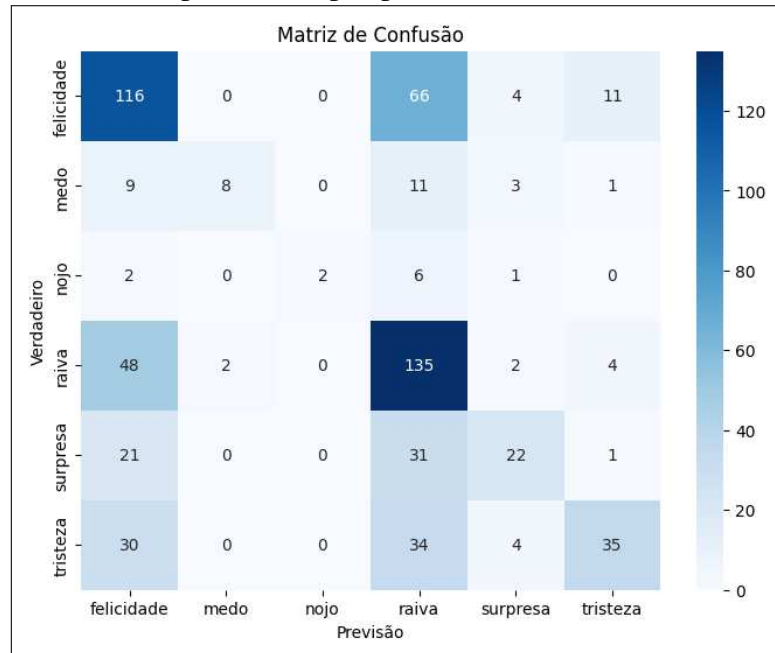
## APÊNDICE A – MATRIZES DE CONFUSÃO DOS MODELOS UNIRRÓTULO

Figura 13 – Matriz de confusão do classificador SVM



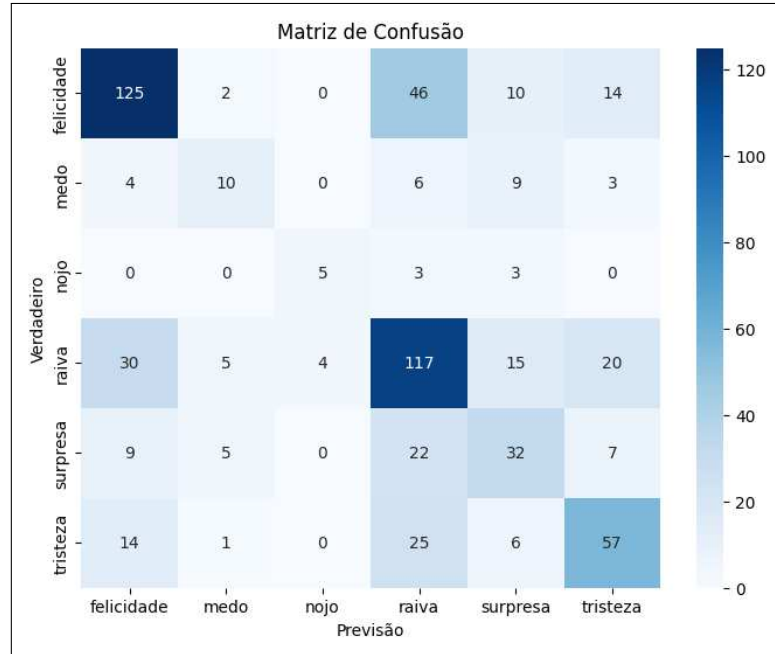
Fonte: elaborado pelo autor (2025).

Figura 14 – Matriz de confusão do classificador SVM: corpus natural pré-processado



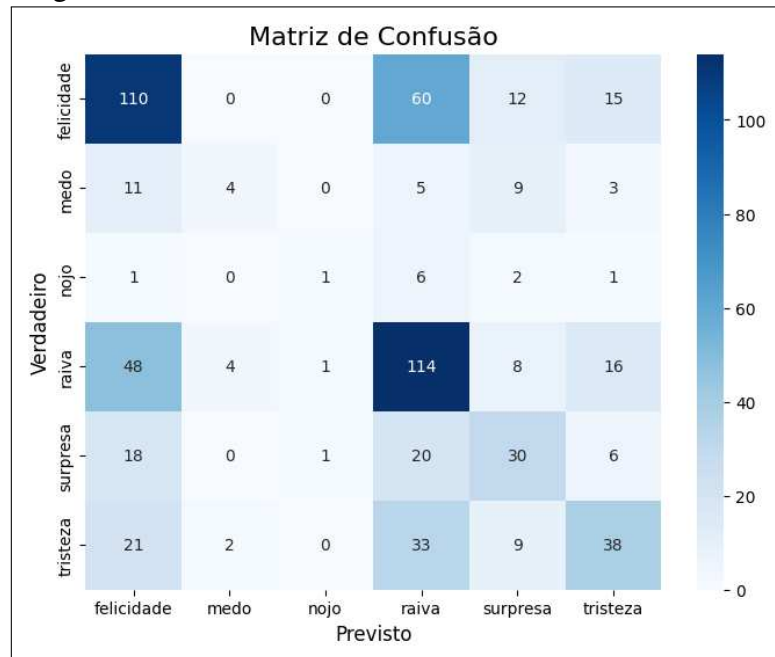
Fonte: elaborado pelo autor (2025).

Figura 15 – Matriz de confusão do classificador SVM: corpus natural pré-processado + sintético



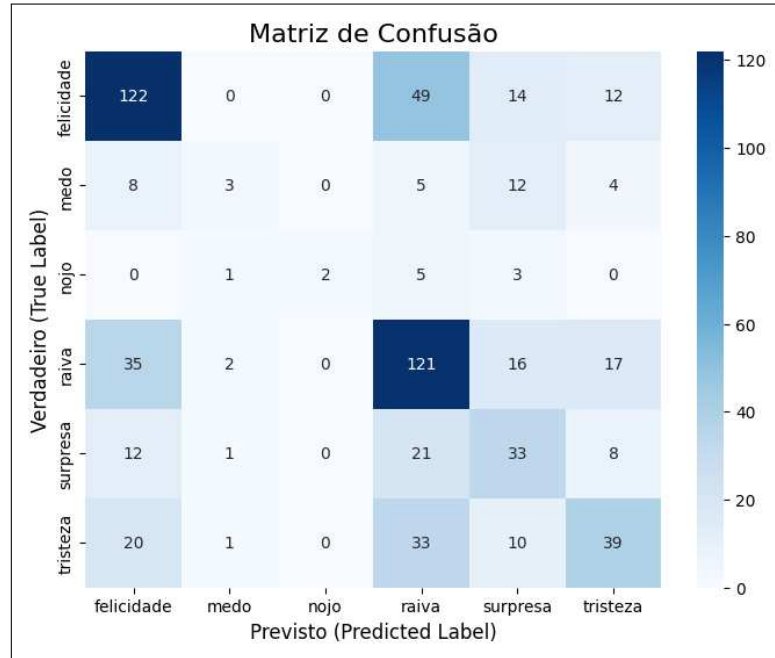
Fonte: elaborado pelo autor (2025).

Figura 16 – Matriz de confusão do classificador RNA



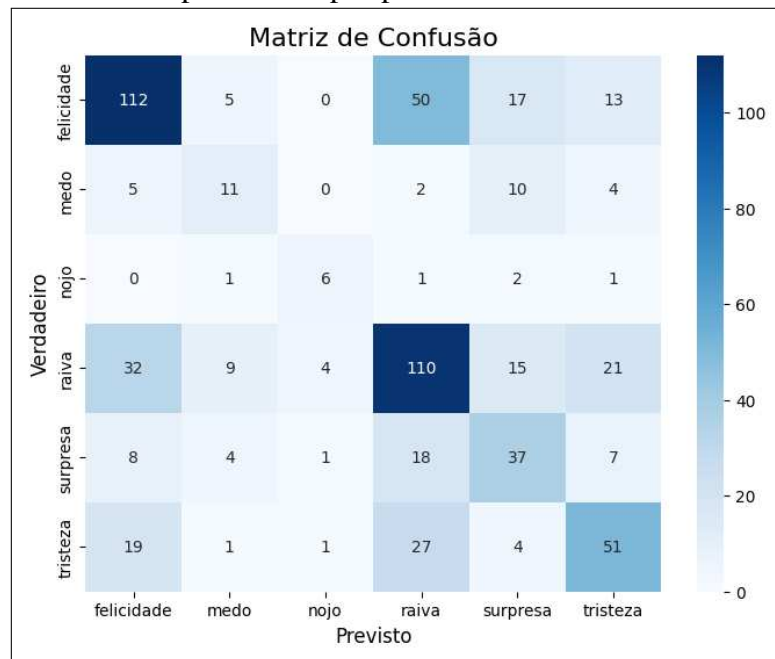
Fonte: elaborado pelo autor (2025).

Figura 17 – Matriz de confusão do classificador RNA: corpus natural pré-processado



Fonte: elaborado pelo autor (2025).

Figura 18 – Matriz de confusão do classificador RNA: corpus natural pré-processado + sintético



Fonte: elaborado pelo autor (2025).