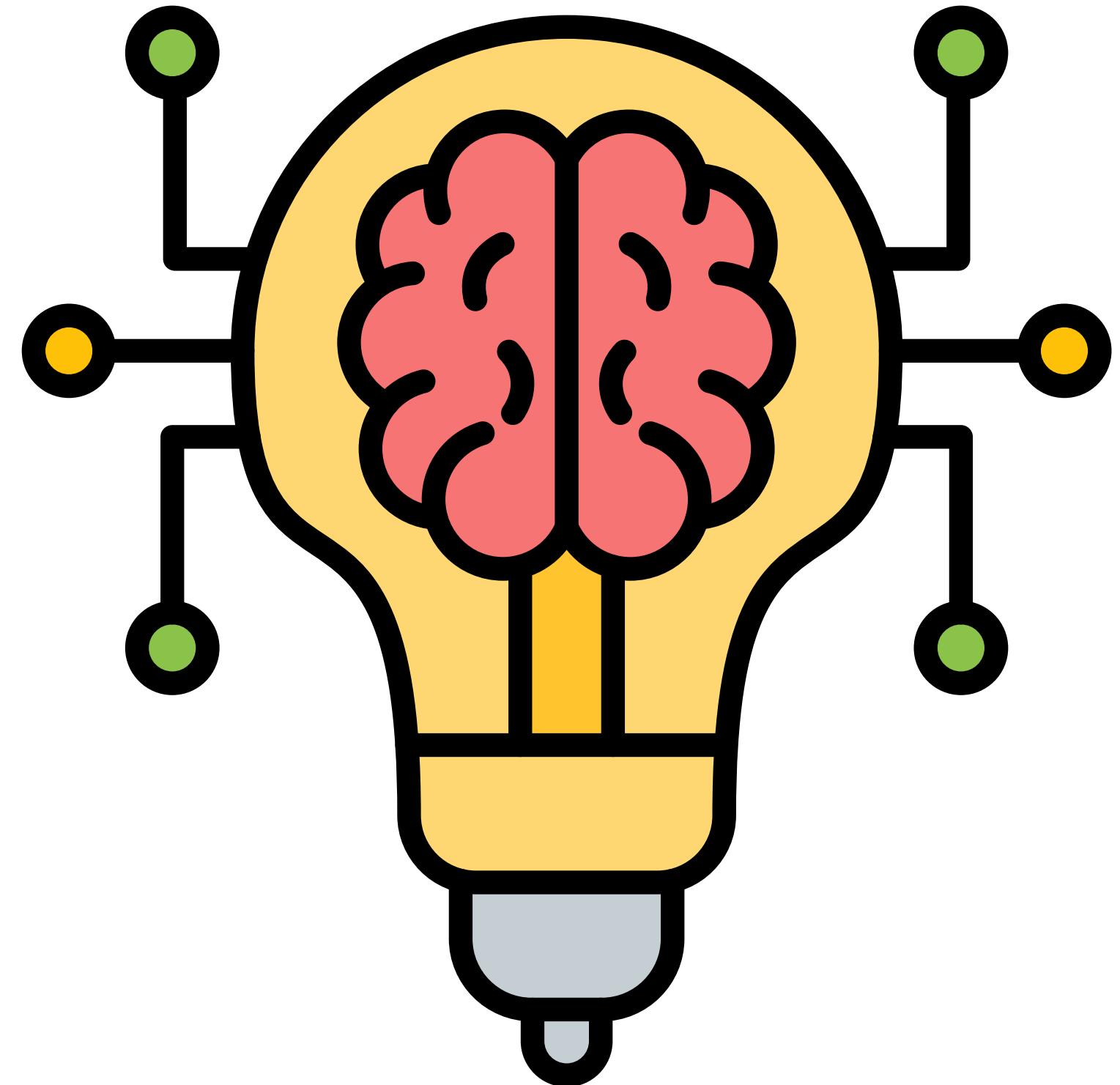


# Predição de avaliação de produtos com Deep Learning

Uma abordagem de NLP para análise de  
reviews e ratings



# Equipe

Carlos Henrique Vila Medeiros  
email: chvm@cin.ufpe.br

Danielle Karla Alves da Silva  
email: dkas@cin.ufpe.br



# Introdução

- O crescimento do comércio eletrônico gera um volume massivo de dados textuais, como avaliações de consumidores, que contêm informações valiosas para empresas como a Amazon.
- As avaliações desempenham um papel crucial na decisão de compra e são utilizadas pelas empresas para moldar estratégias de marketing e na melhoria de produtos.
- A análise manual dessas avaliações enfrenta desafios como o grande volume de dados e a subjetividade nas opiniões dos clientes.



# Introdução

- O Processamento de Linguagem Natural (NLP) automatiza essa análise, permitindo prever notas a partir de avaliações escritas, sendo também possível corrigir inconsistências relacionadas a notas/avaliações.
- Essa abordagem permite otimizar produtos e serviços, além de fornecer insights estratégicos para decisões orientadas por dados.



# Objetivo

Utilizar modelos de classificação utilizando as abordagens Support Vector Machine (SVM) com Bag of Words (BoW), SVM com embeddings, e Bidirectional Encoder Representations from Transformers (BERT) para analisar e classificar dados de reviews e avaliações do **BENGOO Fone de ouvido Pro Gaming para PS4 PS5 PC** disponibilizados na Amazon.



# Base de dados

## Amazon Reviews' 23

A base de dados utilizada foi a base de reviews da Amazon disponibilizada no Hugging Face (<https://huggingface.co/datasets/McAuley-Lab/Amazon-Reviews-2023>).



Screenshot of the Hugging Face dataset page for "Amazon-Reviews-2023".

**Dataset Card:**

- Datasets: McAuley-Lab/Amazon-Reviews-2023
- Languages: English
- Size: 10B< n <100B
- ArXiv: arxiv:2403.03952
- Tags: recommendation, reviews

**Dataset Viewer:** The viewer is disabled because this dataset repo requires arbitrary Python code execution. Please consider removing the `loading_script` and relying on `automated data support` (you can use `convert_to_parquet` from the datasets library). If this is not possible, please open a discussion for direct help.

**Amazon Reviews 2023:**

Please also visit [amazon-reviews-2023.github.io/](https://amazon-reviews-2023.github.io/) for more details, loading scripts, and preprocessed benchmark files.

[April 7, 2024] We add two useful files:

1. `all_categories.txt`: 34 lines (33 categories + "Unknown"), each line contains a category name.
2. `asin2category.json`: A mapping between `parent_asin` (item ID) to its

**Downloads last month:** 10,263

**Models trained or fine-tuned on McAuley...**

- hyp1231/blair-roberta-base
- ashok2216/gpt2-amazon-sentime...
- danzblgn/Sentiment-Analysis-Cu...
- hyp1231/blair-roberta-large

# Produto selecionado



O dataset selecionado inclui as avaliações e notas do BENGOO Fone de ouvido Pro Gaming para PS4 PS5 PC na base de dados da Amazon.



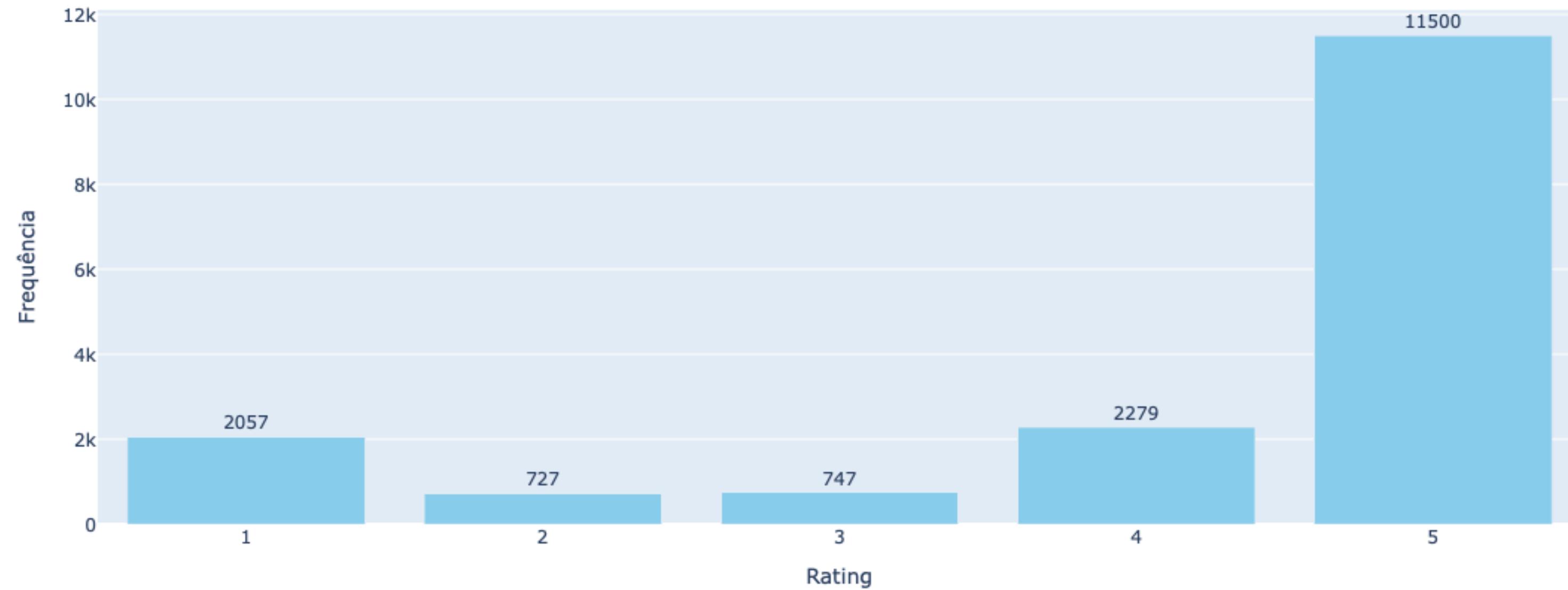
# Pré-processamento

- Selecionamos na base de dados as avaliações referente ao BENGOO Fone de ouvido Pro Gaming para PS4 PS5 PC, pois estava das métricas de mais de 5000 comentários e nota inferior a 4.5.
- O dataframe foi criado com os dados referentes as notas (rating) e avaliações escritas (reviews).
- Os dados de texto foram transformados em número e os sentimentos foram mapeados pelas notas.

• • • • • • • •

# Análises exploratórias

Frequência de Ratings do BENGOO Stereo Pro Gaming Headset for PS4



# Análises exploratórias

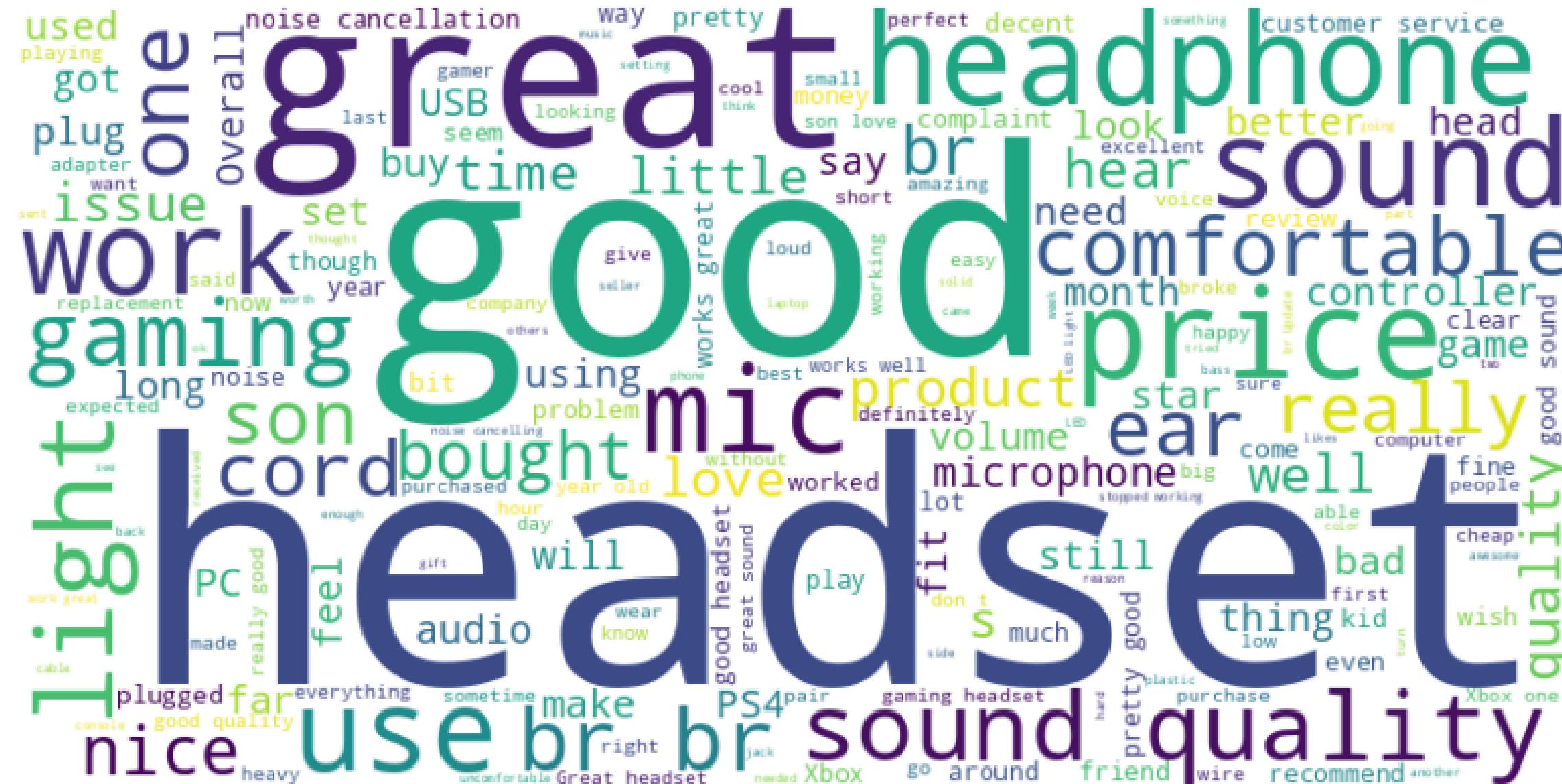
Nuvem de palavras para o rating 5 do BENGOO Stereo Pro Gaming Headset for PS4



# Análises exploratórias

1

Nuvem de palavras para o rating 4 do BENGOO Stereo Pro Gaming Headset for PS4



# Análises exploratórias

Nuvem de palavras para o rating 3 do BENGOO Stereo Pro Gaming Headset for PS4



# Análises exploratórias

3

Nuvem de palavras para o rating 2 do BENGOO Stereo Pro Gaming Headset for PS4



# Análises exploratórias

4

Nuvem de palavras para o rating 1 do BENGOO Stereo Pro Gaming Headset for PS4



# Modelos utilizados

01

SVM + BOW

02

SVM + Embeddings

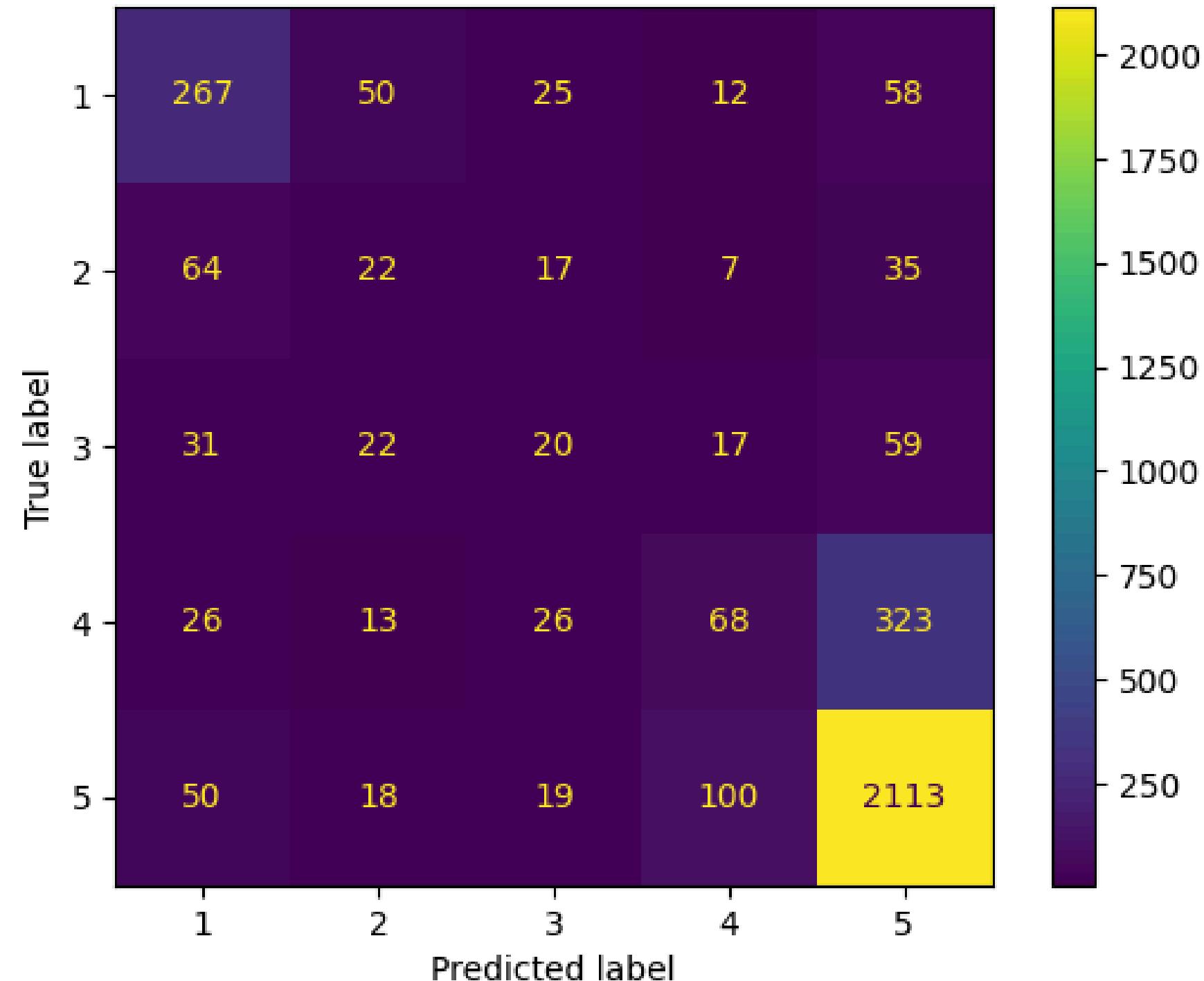
03

BERT



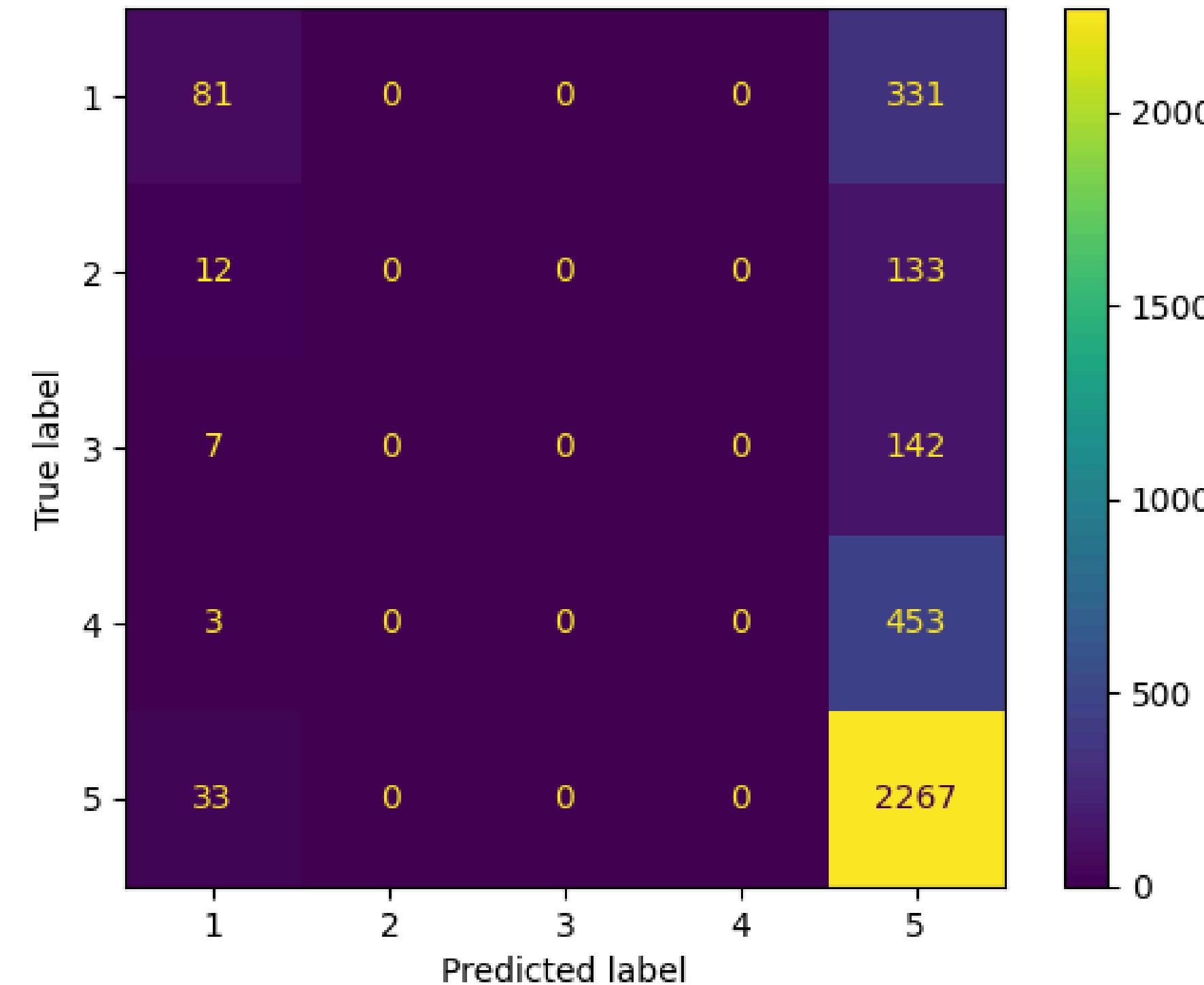
# SVM + BOW

Matriz de Confusão as 5 Classes

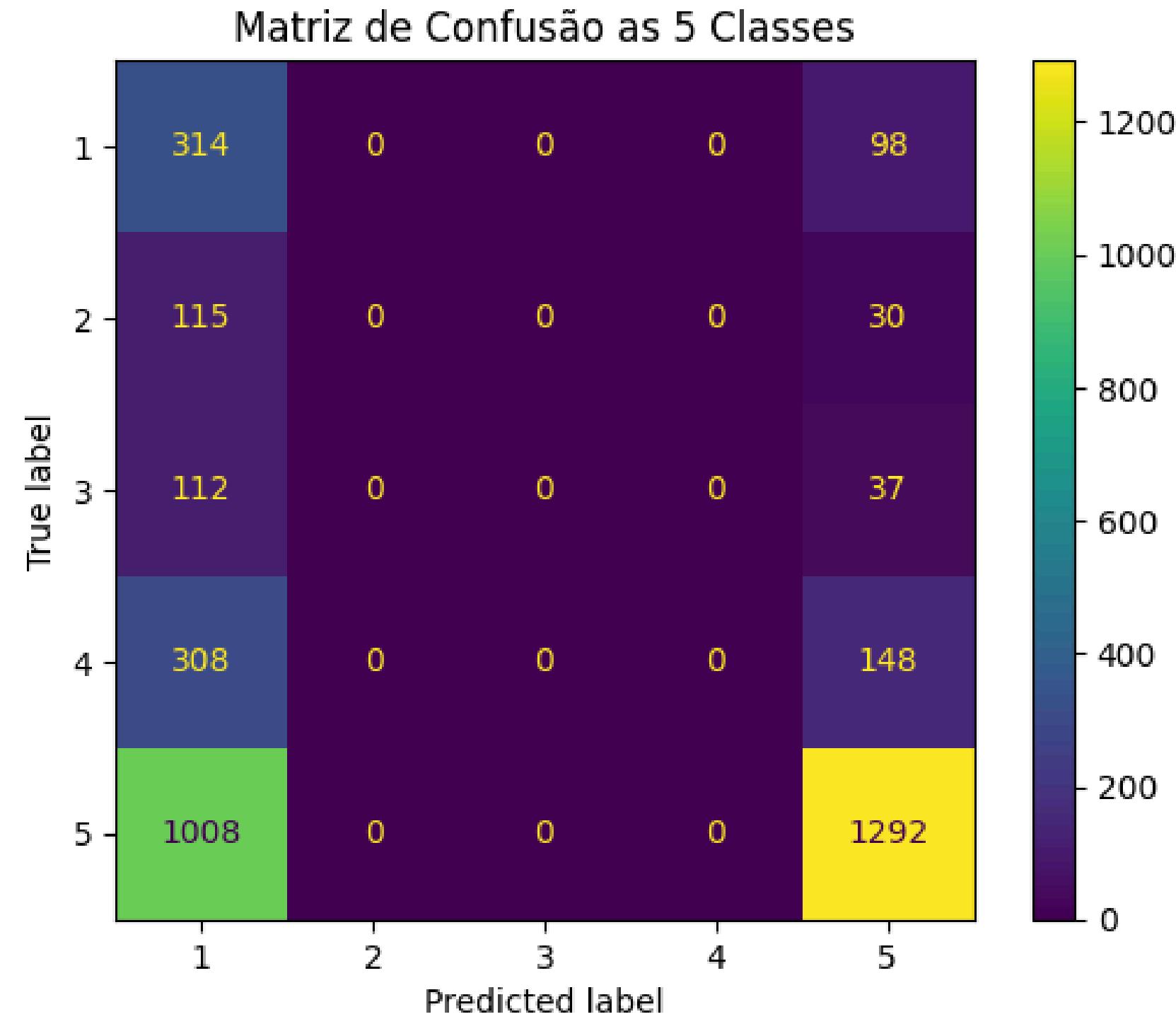


# SVM + Embeddings

Matriz de Confusão as 5 Classes



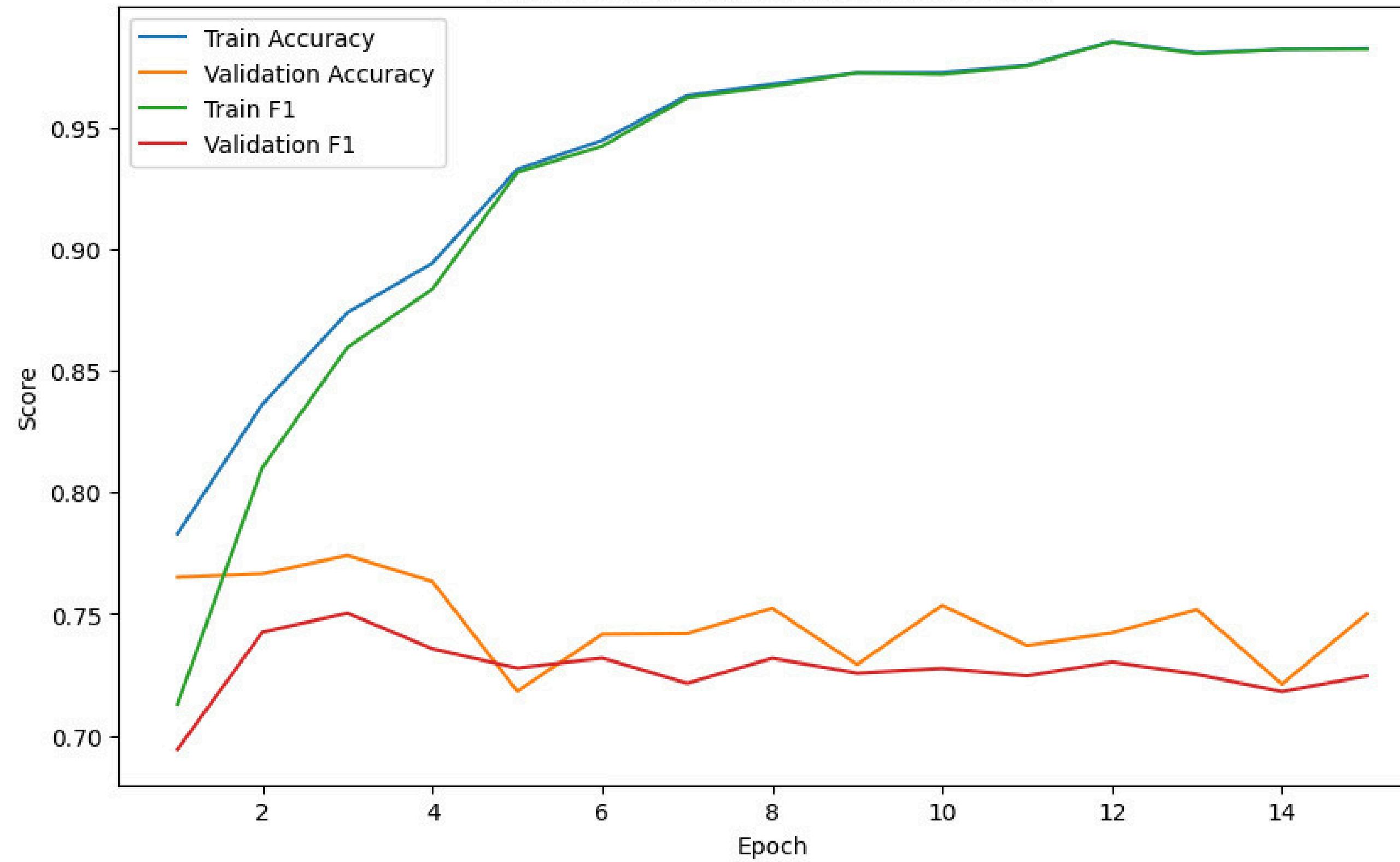
# SVM + Embeddings (Word2Vec)



# BERT

19

BERT Performance Metrics Over Epochs



# In-context learning

20

The screenshot shows a Jupyter Notebook interface with the following details:

- Title:** nlp\_amazon\_headset.ipynb
- Toolbar:** Arquivo, Editar, Ver, Inserir, Ambiente de execução, Ferramentas, Ajuda, Salvo pela última vez às 22:27.
- Connect:** Conectar A100 RAM alta | Gemin

The notebook content is as follows:

```
[ ] import openai

# Configuração da API OpenAI
openai.api_key = "coloque a sua key"

# Função para usar In-Context Learning com o modelo da OpenAI
def classify_review(text):
    # Construção do prompt
    system_prompt = "You are an AI that classifies reviews with ratings from 1 to 5 based on the provided text."
    examples = """
Here are some examples:
- Text: \"The item arrived broken, very poor quality.\" | Rating: 1
- Text: \"Great product, fast delivery, very satisfied!\" | Rating: 5
- Text: \"Average quality, but it works.\" | Rating: 3
"""
    user_prompt = f"Classify the following review:\nText: \"{text}\"\nRating:"

    # Chamada à API com o formato de Chat
    response = openai.chat.completions.create(
        model="gpt-4",
        messages=[
            {"role": "system", "content": system_prompt},
            {"role": "user", "content": examples + user_prompt}
        ],
        temperature=0 # Para maior consistência
    )
```

# In-context learning

```
# Extrai o rating da resposta
return response.choices[0].message.content.strip()

# Testando a função em uma amostra do dataset
sample_text = df_final.iloc[17304]['text']
predicted_rating = classify_review(sample_text)

print(f"Texto: {sample_text}")
print(f"Rating Preditivo: {predicted_rating}")

→ Texto: A little uncomfortable on the ears but still a good product.
Rating Preditivo: 4

[ ] # Testando a função em uma amostra do dataset
sample_text = df_final.iloc[3]['text']
predicted_rating = classify_review(sample_text)

print(f"Texto: {sample_text}")
print(f"Rating Preditivo: {predicted_rating}")

→ Texto: Item received broken poor quality. Adjusting headband cleanly made. Seller asked me to change my review. in exchange for a Possibl
```

# In-context learning

22

The screenshot shows a Jupyter Notebook interface with the following details:

- Title:** nlp\_amazon\_headset.ipynb
- Toolbar:** Arquivo, Editar, Ver, Inserir, Ambiente de execução, Ferramentas, Ajuda, Salvo pela última vez às 22:27.
- Cell 1:** Contains Python code for testing a function on a dataset sample:

```
[ ] # Testando a função em uma amostra do dataset
sample_text = df_final.iloc[3]['text']
predicted_rating = classify_review(sample_text)

print(f"Texto: {sample_text}")
print(f"Rating Preditivo: {predicted_rating}")
```

Output:

```
Texto: Item received broken poor quality. Adjusting headband cheaply made. Seller asked me to change my review, in exchange for a Possible Rating Preditivo: 1
```
- Cell 2:** Displays the `df_final` DataFrame:

	rating	text
0	5	Great
1	5	Kids loved them
2	5	ok
3	1	Item received broken poor quality. Adjusting h...
4	5	If you're really into gaming I should have thi...
...	...	...
17305	5	These are awesome with my ps4!! FYI: need a fe...
17306	4	I love this headset it has great sound quality...
17307	4	A little uncomfortable on the ears but still a...
17308	5	Great headset
17309	5	My daughter loves the headset for playing Fort...

17307 rows × 2 columns

# Comparando os modelos treinados

Métricas	SVM + BOW	SVM + EMB	BERT
ACC / VAL	0.7192	0.6782	0.7501
F1 / VAL	0.6898	0.5705	0.7246

• • • • • • •



# Conclusões

- Comparando os 3 modelos, é possível concluir que o BERT se mostrou mais adequado ao conjunto de dados analisado apresentando melhores valores de acurácia e F1.
- Possivelmente, o BERT lida melhor com dados desbalanceados, como é o caso desse conjunto de dados.

• • • • • • • •



# Perspectivas e trabalhos futuros

- Incorporar outras variáveis e/ou outros tipos de dados (p.ex: imagens) em análises futuras para complementar a classificação textual e fornecer uma visão mais completa. Como também realizar um balanceamento dos dados.
- Para melhorar a acurácia do conjunto de dados de teste é possível aumentar o contexto semântico e ajustar hiperparâmetros.

• • • • • • • •



# Obrigado!

Que no próximo ano, possamos continuar decodificando desafios, treinando modelos e construindo conexões — seja na linguagem natural ou na linguagem da vida.  
Feliz Ano Novo!

Frase criada no ChatGPT

