Análise de sentimentos em tweets

Livia Tanaka
Insper - Instituto de Ensino e Pesquisa
São Paulo, Brasil
liviat1@al.insper.edu.br

I. Dataset

A análise de sentimentos é um processo automático para determinar a polaridade de sentimento de um texto. Essa técnica tem um valor comercial pautado no fato de que a partir dela é possível ter uma visão geral do público sobre determinado assunto a partir de tweets [1]. Assim, possibilitando a empresa a estudar as opiniões de seu público e analisar a satisfação do consumidor. No estudo a seguir, será utilizado um classificador para determinar a classe de cada um dos tweets do dataset Processed twitter sentiment Dataset — Added Tokens [2], como positivo ou negativo.

II. Pipeline de Classificação

A. Pré-processamento

O pré-processamento da base de dados foi efetuado através dos seguintes passos:

1) Transformar todas as letras em minúsculas;
2) Retirar os usernames (palavras que começam com "@") e os links (palavras que começam com "http"), posto que esses não agregam valor positivo ou negativo ao tweet [1];
3) Quando há mais de 3 repetições seguidas de determinada letra, foi retirada as repetições, deixando apenas uma ocorrência (ex. "boooooored" ficou "bored");
4) Substituir todas as palavras pelos seus lemmas, ou seja, a forma como a palavra seria encontrada em um dicionário, sem suas flexões.
5) Retirar stopwords;

Assim, agrupando palavras de mesmo significado morfológico e, portanto, diminuindo o número de features.

B. Pipeline

O modelo escolhido foi o bag-of-words (BoW), pois assume-se que, para esse caso, a ordem das palavras não é tão relevante. O mais importante é o significado de cada uma delas em relação a se é uma palavra com sentido positivo ou negativo. Para compor a pipeline, foi utilizado um vetorizador e um classificador. O vetorizador utilizado foi o CountVectorizer do scikit-learn com o *max features* como 5000 , ele transforma os textos em vetores númericos contando o número de ocorrência das palavras. Já o classificador foi utilizado o método de regressão linear [3], no qual os coeficientes representam os pesos de cada uma das palavras, sendo essas as features para determinar se o texto possui a classe positiva (1) ou negativa (0). Para regressão, foi utilizado *C* como 0.1 e o *max iter* como 1000.

III. Avaliação do classificador

Para avaliar a pipeline montada, foram realizados 100 testes com diferentes partições de treino e teste. A média da acurácia balanceada do modelo foi de aproximadamente 75% com um desvio padrão de 0.02.

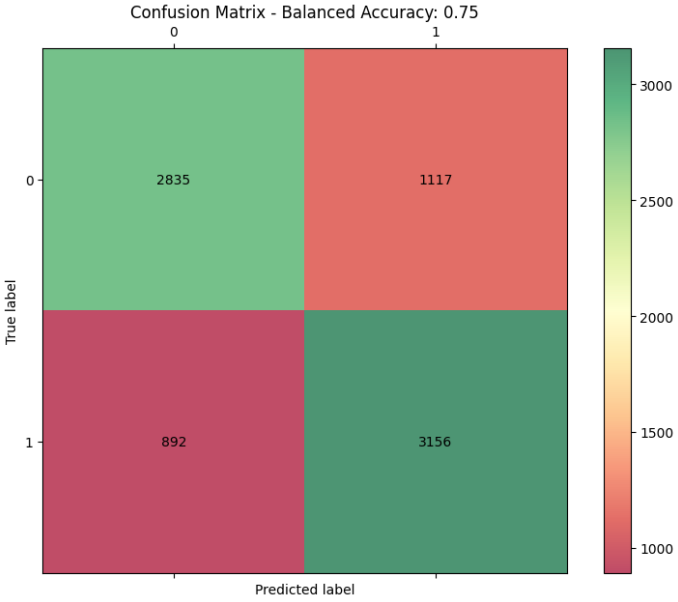

Fig. 1. Matriz de confusão

Na Fig. 1 é possível notar que há uma pequena diferença de resultado, posto que a classificação de textos como negativos tende a ter mais erros do que a de positivos. Uma hipótese para essa diferença é que textos com ironia, normalmente possuem um sentimento negativo expressado por palavras positivas. Por exemplo, frases como "What a great day to be run over by a bus!" e "I love when my phone autocorrects 'my name to 'smartass'" são classificadas como positivas.

A. Palavras mais importantes

Na tabela I, temos as 10 palavras mais importantes para a classificação, tanto positiva, quanto negativa. As palavras negativas estão associadas a sentimentos de tristeza, saudade e dor.Já as positivas são, em geral, sentimentos de agradecimento e onomatopeias de risada e comemoração.

TABLE I
TOP 5 PALAVRAS POSITIVAS E NEGATIVAS

Palavras Positivas	**Valor**
thanks	1.448671
thank	1.343958
welcome	1.175617
yay	1.122639
hehe	1.002394
Palavras Negativas	**Valor**
sad	-2.340818
wish	-1.708286
sick	-1.560927
miss	-1.546740
hurt	-1.406317

IV. ANÁLISE DO DATASET

Tendo em vista que o tamanho do dataset influencia na acurácia do classificador na partição de teste. Na Fig. 2 é possível notar a forma como a acurácia aumenta conforme o tamanho da amostra cresce.

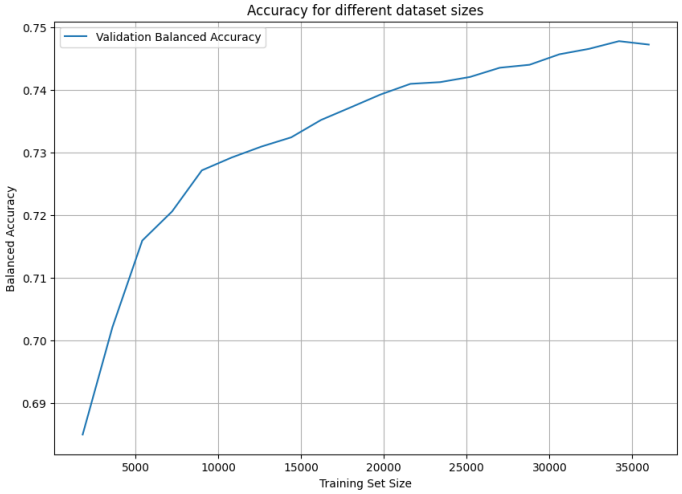

Fig. 2. Acurácia em diferentes tamanhos de dataset

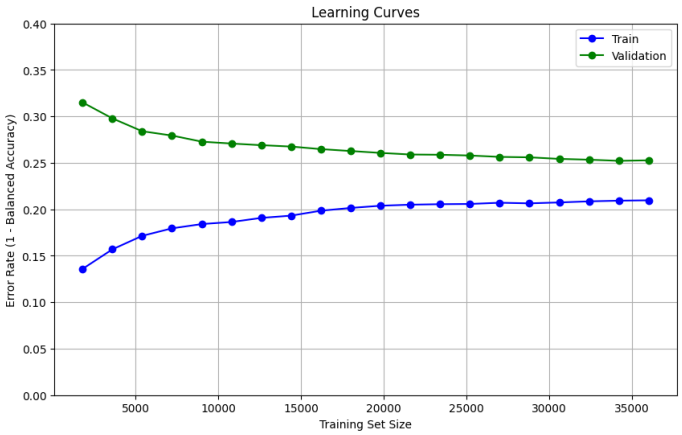

Fig. 3. Taxa de erro em diferentes tamanhos de dataset

Entretanto, esse crescimento tem um limite, denominado como limite de Bayes, no qual a acurácia no conjunto de treinamento se iguala a do conjunto de teste.Na Fig. 3 é possível notar que as taxas de erro (1- acurácia) do treinamento e do teste estão se aproximando, conforme o aumento do dataset. Porém a partir dos 20.000 tweets, a velocidade da aproximação passa a diminuir.

Dessa maneira, pode-se concluir que mesmo havendo espaço para aperfeiçoamento do modelo com um maior número de dados, é preciso ponderar se o dinheiro e tempo necessários para categorizar mais tweets compensa o ganho de performance que será obtido com esse aumento dos dados.

V. ANÁLISE DE TÓPICOS

Para a análise de tópicos, foi construido uma pipeline com três camadas sendo a primeira o vetorizador, a segunda um NMF (Non-Negative Matrix Factorization) para modelar os tópicos e a terceira o classificador. Foi testada a modelagem com diferentes números de componentes (de 2 a 8) e a melhor acurácia foi 57% com 6 componentes. Ambos os tópicos possuiam palavras positivas ou neutras, o que pode ter sido a causa de uma performance pior que a da pipeline anterior. Além disso, o fato de que uma maioria desproporcional dos tweets tem maior influência do tópico 1, como visto na Fig. 4

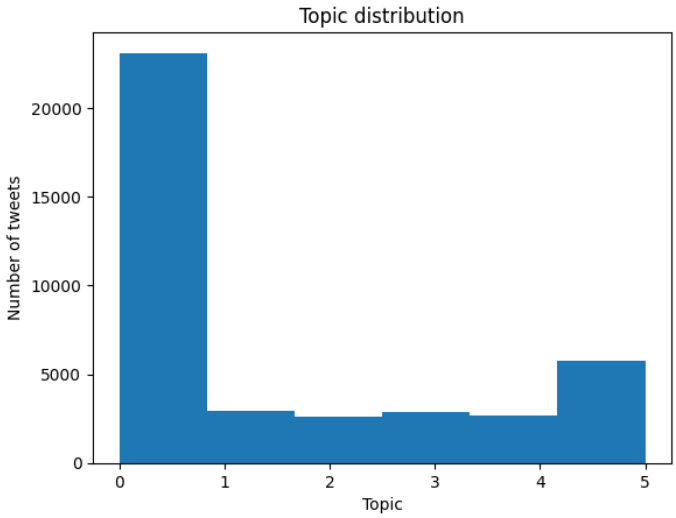

Fig. 4. Distribuição dos tópicos entre os tweets

REFERENCES

[1] G. Alec, “Twitter Sentiment Classification using Distant Supervision.”, 2009.
[2] Halemo GPA. (2024). Halemo GPA. (2024). Processed twitter sentiment Dataset — Added Tokens [Data set]. Kaggle. https://doi.org/10.34740/KAGGLE/DS/5568348
[3] J. Daniel and J. Martin, “Speech and Language Processing,” Jan. 2023. Available: https://web.stanford.edu/ jurafsky/slp3/5.pdf