NOVO ENSINO SUPLEMENTAR

ANÁLISE DE SENTIMENTOS A PARTIR DE ALGORITMOS DE CLASSIFICAÇÃO MULTICLASSES

ESTUDANTE: RUAN LIMA
ORIENTADOR: JULIANO GENARI

INTRODUÇÃO

No contexto de Machine Learning, a tarefa de classificação pode ser abordada de diversas maneiras, cada uma com suas próprias vantagens e desvantagens. Neste projeto, implementaremos uma variedade de algoritmos de classificação multiclasses para identificar o modelo mais eficaz na análise de sentimentos em textos provenientes de redes sociais.

Ao final deste projeto, esperamos não apenas identificar o melhor modelo de classificação, mas também obter uma compreensão mais profunda sobre como diferentes técnicas de vetorização de texto e algoritmos de Machine Learning interagem e influenciam a precisão da análise de sentimentos.

DATASET

O dataset é formado por linhas de dados obtidas em redes sociais. Nelas, contém informações como o conteúdo principal, o sentimento descrito e o texto obtido. Nele são caracterizados quatro tipos de sentimentos: Positive, Negative, Neutral e Irrelevant. O dataset contém todas as informações em inglês. Abaixo, observamos um exemplo das linhas deste dataset.

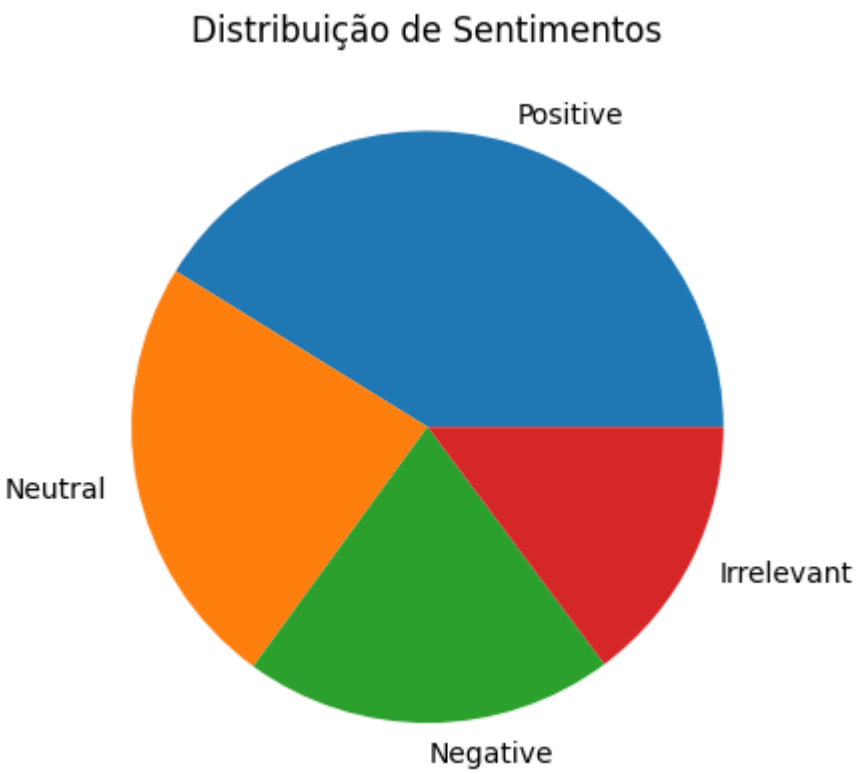
	none	none.1	Sentiment	Text
0	2401	Borderlands	Positive	im getting on borderlands and i will murder yo...
1	2401	Borderlands	Positive	I am coming to the borders and I will kill you...
2	2401	Borderlands	Positive	im getting on borderlands and i will kill you ...
3	2401	Borderlands	Positive	im coming on borderlands and i will murder you...
4	2401	Borderlands	Positive	im getting on borderlands 2 and i will murder ...

ANÁLISE DOS DADOS

O nosso conjunto de dados inicialmente precisa de um tratamento. Mais especificamente com linhas vazias e elementos com o tipo diferente de str. Utilizando a biblioteca $pandas$ foram identificadas e removidas 858 linhas com elementos nulos. Além disso, 17 linhas com conteúdos cujo tipo era diferente de str foram removidas.

Utilizando a biblioteca $matplotlib$ conseguimos visualizar que os nossos dados estão balanceados. Veja abaixo:

$Sentimento\ Positive$: 20832
$Sentimento\ Neutral$: 18318
$Sentimento\ Negative$: 22542
$Sentimento\ Irrelevant$: 12990



PRÉ-PROCESSAMENTO DOS DADOS

Como estamos trabalhando com textos, esta etapa é crucial. Nela, iremos remover toda e qualquer informação desnecessária para o nosso projeto. Iremos remover acentos, pontuações caracteres especiais e padronizar os nossos textos para $lowercase$. Além disso, faremos alguns processos como tokenização, remoção de stop words, stemização e lematização. Apenas depois de ter feito tudo isso poderemos seguir com o projeto.

METODOLOGIA

Iremos utilizar algoritmos como: K-Nearest Neighbors (KNN), Decision Tree, Support Vector Machine (SVM), Naive Bayes e Random Forest. Todos os algoritmos utilizados pertencem à biblioteca $sklearn$. Além disso, este projeto explora duas técnicas populares de vetorização de texto: TF-IDF (Term Frequency-Inverse Document Frequency) e Bag of Words (BoW). A combinação dessas técnicas com diferentes algoritmos de classificação permitirá uma avaliação abrangente de sua eficácia na tarefa de análise de sentimentos. Com um conjunto de dados balanceado e volumoso, a $acurácia$ será utilizada como a principal métrica de desempenho dos modelos. Utilizaremos bibliotecas como o $numpy$ e $matplotlib$ para a visualização dos resultados.

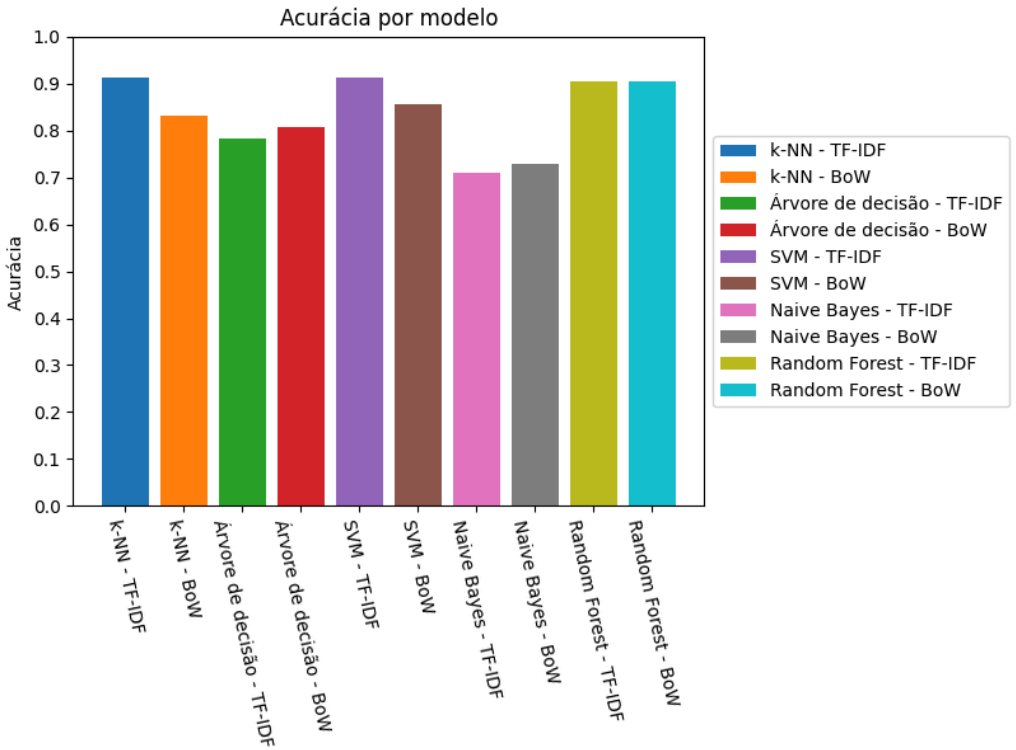
Com isso, implementaremos os nossos algoritmos e faremos o cruzamento com os dois tipos de vetorização. Veja abaixo um exemplo:

RESULTADOS

Após a definição, treinamento e predição de cada modelo, obtemos os seguintes resultados:

Modelo	Acurácia
KNN + TF-IDF	91%
KNN + BoW	83%
Decision Tree + TF-IDF	78%
Decision Tree + BoW	81%
SVM + TF-IDF	91%
SVM + BoW	86%
Naive Bayes + TF-IDF	71%
Naive Bayes + BoW	73%
Random Forest + TF-IDF	91%
Random Forest + BoW	90%

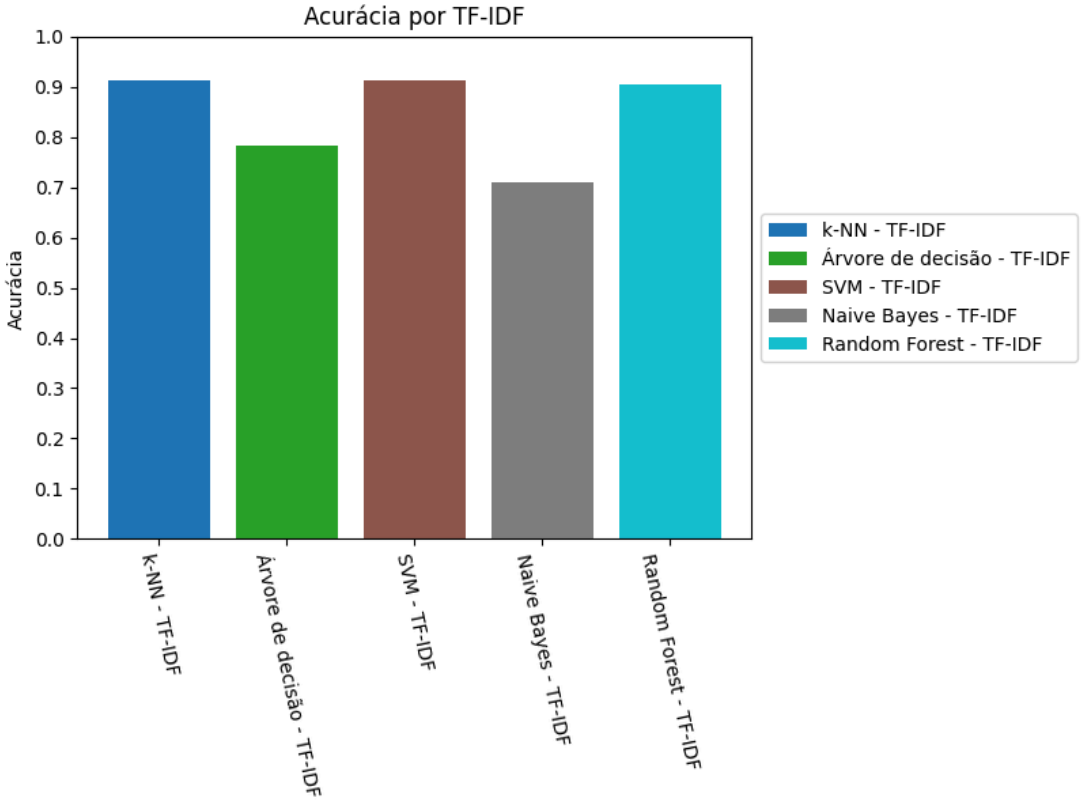
Abaixo podemos visualizar esses resultados graficamente:



Os três modelos com a melhor acurácia foram:

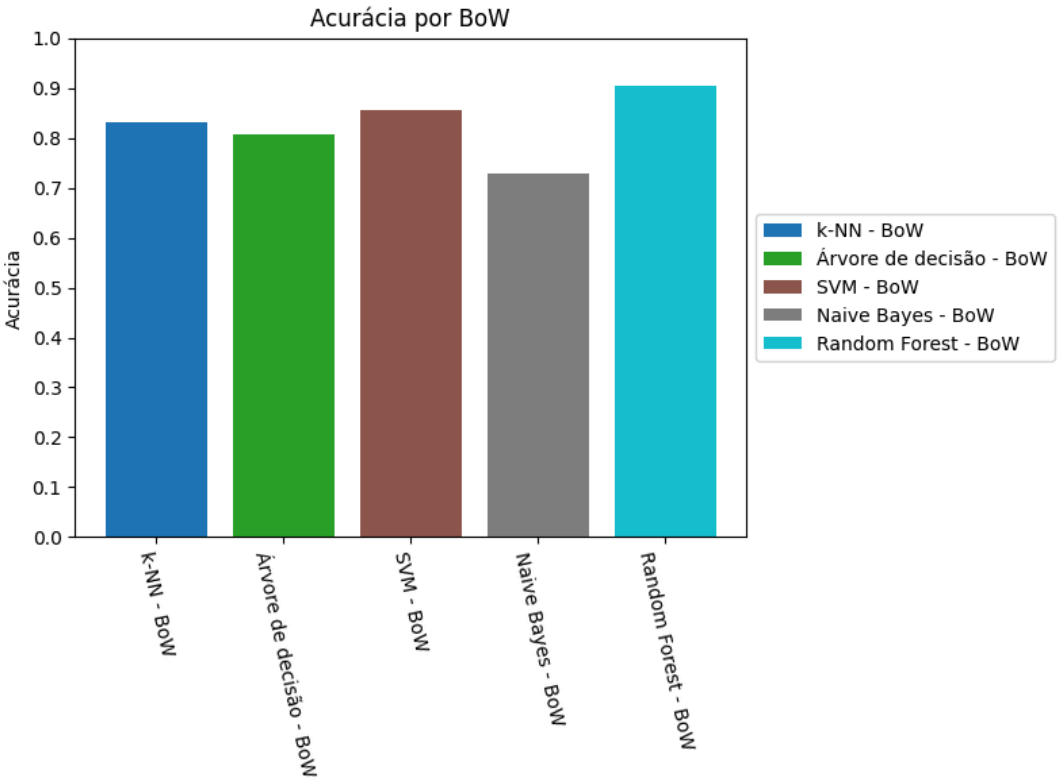
$$KNN + TFIDF$$
$$Random\ Forest + TFIDF$$
$$SVM + TFIDF$$

Também podemos olhar para o desempenho dos modelos cuja vetorização foi o TF-IDF:



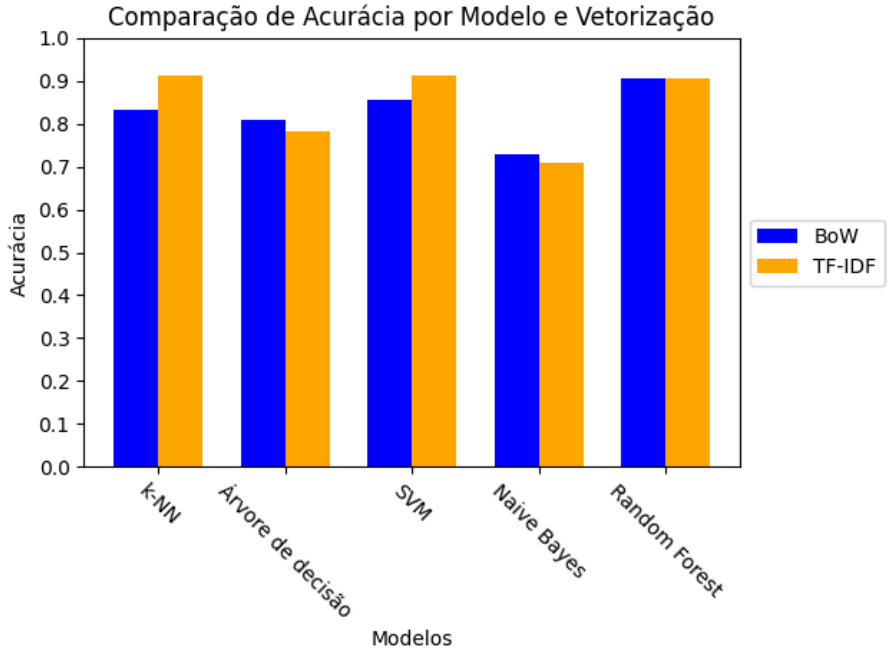
Os melhores foram o KNN, SVM e $Random\ Forest$.

E o desempenho dos modelos com o Bag of Words:



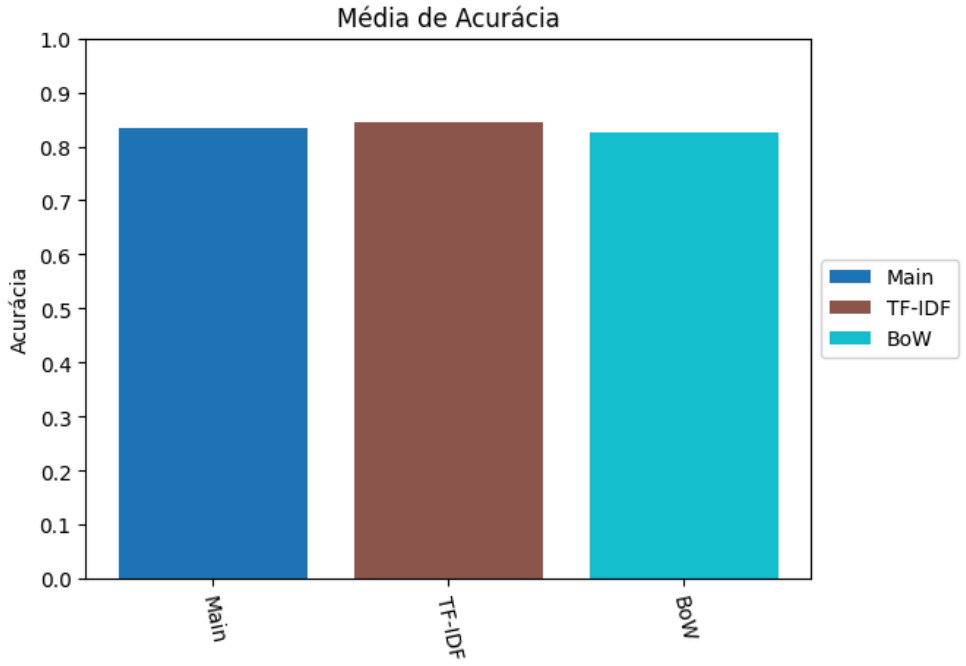
Os melhores também foram o $KNN,\ SVM\ e\ Random\ Forest$

Abaixo podemos visualizar o desempenho de cada algoritmo com cada vetorização:

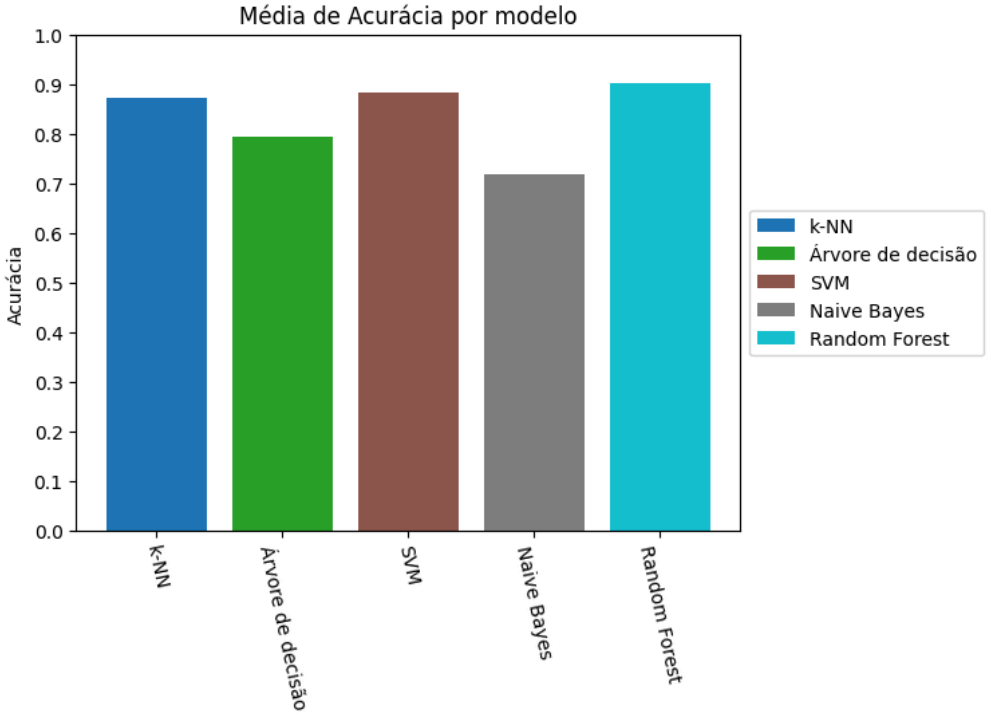


O TF-IDF foi superior na maioria dos modelos. Além disso, a maior diferença entre os dois tipos pode ser observada no KNN, enquanto nos outros algoritmos a diferença é mínima.

De modo geral, a média de acurácia de todos os algoritmos, apenas TF-IDF e apenas BoW foi bem semelhante.



Média de acurácia por cada modelo:



Com base nos resultados e gráficos acima podemos concluir com propriedade que os melhores algoritmos foram o KNN, SVM e o $Random\ Forest$, sendo estes, os que obtiveram as melhores acurácias. O $Naive\ Bayes$ se mostrou o mais ineficiente nesta tarefa. E o Bag of Words a vetorização mais ineficiente

Os resultados são conclusivos e satisfatórios. Através da implementação e avaliação de diversos algoritmos de classificação multiclasses, este projeto identificou o Random Forest como o modelo mais eficaz para a tarefa de análise de sentimentos em textos de redes sociais.

Os resultados confirmam que, para conjuntos de dados balanceados e volumosos, o Random Forest não só proporciona robustez e estabilidade, mas também uma capacidade superior de generalização.

Além disso, concluímos que a vetorização TF-IDF é mais eficiente em capturar a relevância dos termos em diferentes documentos, contribuindo para a precisão das previsões de sentimentos.

AGRADECIMENTOS

Agradecimento especial ao $specionate\ nlp$ pela coleta e disponibilização dos dados, e ao sklearn pela disponibilização das ferramentas utilizadas no projeto, sendo extremamente necessários para este projeto.

REFERÊNCIAS

Specionate, (2020). Twitter Sentiment Analysis.
https://www.kaggle.com/datasets/jp797498e/twitter-entity-sentiment-analysis
[Acesso em Julho de 2024].

Genari J., (2022). Sarcastic headlines classification.
https://nanogennari.medium.com/sarcastic-headlines-classification-9738b1541229
[Acesso em Julho de 2024].