FACULDADE DE TECNOLOGIA DA BAIXADA SANTISTA – FATEC RUBENS LARA

CURSO DE CIÊNCIA DE DADOS

ANÁLISE DE REVIEWS DA STEAM UTILIZANDO TF-IDF

Aluno: Caio Kenji de Paula Maeshiro

Disciplina: Álgebra Linear

Período: 2º semestre de 2025

Santos – SP

2025

Descrição do Dataset

O projeto tem como o objetivo a análise de um dataset composto por avaliações públicas da plataforma Steam. O dataset utilizado contém mais 6,4 milhões de reviews em inglês, disponibilizados de forma pública e anônima, reunindo informações sobre a opinião de usuários a respeito de diversos jogos disponíveis na plataforma.

Cada registro do conjunto de dados inclui as seguintes colunas:

- **Review text:** o texto livre da avaliação feita pelo usuário;
- **Game ID:** o identificador numérico do jogo ao qual a avaliação pertence;
- **Sentiment:** o sentimento da review, podendo ser classificado como positivo ou negativo;
- **Helpful:** número de usuários que consideraram aquela avaliação útil.

O arquivo original foi disponibilizado em formato CSV compactado. Devido ao seu grande volume de informações, totalizando mais 6 milhões de registros. Para fins de processamento e viabilidade computacional, foi considerado apenas uma amostra de 100.000 de registros, extraída a partir do dataset completo. A extração foi feita no script chamado "Filtro" e a amostra foi salva em um novo arquivo denominado "reviews_filtradas.csv", que serviu de base para as etapas seguintes.

A partir dessa amostra, foi realizado um filtro específico para o jogo de ID 10180, que corresponde ao título Call of Duty: Modern Warfare 2 (2009). Essa filtragem teve como finalidade concentrar o estudo em um único jogo bastante avaliado pelos usuários.

Tema do Projeto

Este trabalho se dedica fazer a análise de avaliações sobre o jogo Call of Duty: Modern Warfare 2 (2009), buscando descobrir as semelhanças entre a primeira avaliação registrada e as subsequentes. Para alcançar este objetivo, empregar-se o método TF-IDF (Term Frequency – Inverse Document Frequency), que transforma textos em dados numéricos com base na importância das palavras.

Com essa representação vetorial, foi possível calcular o grau de similaridade entre as avaliações por meio da métrica de similaridade do cosseno, identificando quais reviews apresentam conteúdos mais próximos entre si.

Etapas Realizadas

```python
import pandas as pd

df = pd.read_csv("steam.csv", sep=',', decimal=',')
df = df.head(1000000)
df.to_csv("reviews_filtradas.csv", index=False, sep=',', decimal=',')
```

Inicialmente, realizou-se a redução do tamanho do conjunto de dados original para facilitar a análise. O arquivo original, lido através da biblioteca Pandas, era extenso, então optou-se por pegar uma amostra, pegando as primeiras 100.000 linhas. Essa seleção foi armazenada em um novo arquivo, nomeado "reviews_filtradas.csv", que serviu como base para as fases seguintes do processo.

```python
import pandas as pd
import re
import nltk
from nltk.corpus import stopwords
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import cosine_similarity

#Limpando o dataset
df = pd.read_csv("reviews_filtradas.csv", sep=',', decimal=',')
df = df.drop(columns=["1", "0"])
df = df.rename(columns={"10": "id_jogo", "Ruined my life.": "Review"})
df = df[df["id_jogo"] == 10180]
df["Review"] = df["Review"].astype(str)
df = df.drop_duplicates(subset='Review')
df = df.reset_index(drop=True)
print(df.head())
```

Na sequência, os dados do arquivo passaram por um processo de leitura, limpeza e filtragem. Primeira, algumas colunas que não agregava ao projeto foram descartadas, enquanto outras receberam novos nomes para facilitar o entendimento do conjunto de dados. Em seguida, aplicou-se um filtro para isolar os dados do jogo com ID 10180, correspondente ao título Call of Duty: Modern Warfare 2 (2009), focando a análise nas opiniões sobre esse título específico.

Após o filtro, todas as avaliações foram convertidas para o tipo texto, garantindo consistência na manipulação dos dados. Também foram removidas eventuais duplicatas, de modo a evitar que repetições influenciassem nos resultados. Por fim, o índice do conjunto de dados foi redefinido, resultando em um DataFrame limpo.

```
#Padronizando
nltk.download("stopwords")
def limpar_review(review):
    review = review.lower()
    review = re.sub(r"&[a-z]+;", " ", review)
    review = re.sub(r"[^a-z\s]", "", review)
    palavras = review.split()
    palavras = [p for p in palavras if p not in stopwords.words("english")]
    return " ".join(palavras)
df["Review_limpa"] = df["Review"].apply(limpar_review)
print(df[["Review", "Review_limpa"]].head())
```

Depois da filtragem de dados, foi realizado o tratamento do texto, buscando uniformizar e aprontar as avaliações para a análise quantitativa.

Inicialmente, as stopwords da biblioteca NLTK (Natural Language Toolkit), que são palavras muito comuns no inglês, foram carregadas e ajustadas. Essas palavras foram removidas para que a análise se concentrasse nas palavras-chave das opiniões dos usuários.

Em seguida, foi criada a função "limpar_review". Essa função executa uma série de transformações:

- **Converte todo o texto para letras minúsculas** garantindo uniformidade entre palavras semelhantes escritas de maneiras diferentes;
- **Remove símbolos e caracteres especiais** como pontuações, números e códigos HTML;;
- **Divide o texto em palavras isoladas** possibilitando o tratamento individual de cada termo;
- **Elimina as stopwords** mantendo apenas as palavras de maior relevância semântica;
- **Reagrupa o texto limpo** em uma nova forma padronizada.

O resultado desse processo foi colocado em uma nova coluna chamada Review_limpa, que contém a versão processada de cada avaliação original. Essa coluna serviu de base para as etapas seguintes de vetorização e cálculo de similaridade.

```
#Trasnformando os textos em valores numéricos
vetorizador = TfidfVectorizer()
reviews_matrix = vetorizador.fit_transform(df["Review_limpa"])
indice = 0
similaridade = cosine_similarity(reviews_matrix[indice], reviews_matrix)
similaridades = list(enumerate(similaridade[indice]))
similaridades = sorted(similaridades, key=lambda x: x[1], reverse = True)
```

Com os textos já limpos, foi aplicada a técnica TF-IDF. A implementação foi feita utilizando o recurso TfidfVectorizer() da biblioteca scikit-learn, resultando em uma matriz esparsa, designada como reviews_matrix.

A seguir, empregou-se a função cosine_similarity() para quantificar o nível de similaridade entre as diversas análises, com foco na primeira avaliação como a base de comparação.

Os resultados foram então colocados em ordem decrescente, o que facilitou a identificação de avaliações com conteúdo bastante similar à avaliação inicial, revelando tendências e assuntos recorrentes nas impressões dos jogadores.

```
#Apresentando resultados
print(f"\nBase Review 0: {df['Review_limpa'][0]}")
print(f"\nTop 10 reviews mais parecidas:")
for i, score in similaridades[1:11]:
    print(f"\nReview {i}: \n{df["Review_limpa"][i]} \n(similaridade: {score:.2f})")
```

Por fim, o código exibe no terminal a review base utilizada como referência e as dez avaliações mais semelhantes encontradas pelo cálculo de similaridade do cosseno. O laço for percorre os índices e valores de similaridade armazenados na lista similaridades, apresentando para cada item o texto correspondente e o respectivo grau de semelhança numérica.

Análise dos Resultados

Após empregar o método TF-IDF, calculou-se a similaridade de cosseno entre as opiniões sobre o jogo Call of Duty: Modern Warfare 2 (2009). A intenção era descobrir quais análises mostravam uma ligação semântica mais forte com a primeira análise (posição 0).

A avaliação base selecionada pelo código foi:

Base Review 0:

“makarov thought saw ghost got spooked dropped soap soon paid price”

Resultado das 10 avaliações mais semelhantes

Posição	Índice	Trecho da Review	Similaridade
1	2961	ive played almost every call duty title... selling soap...	0.24
2	1492	revenge ghost roach die soap price kill shepherd lol	0.24
3	2681	better ghost	0.23
4	1759	dont drop soap price decrease	0.20
5	1129	rip ghost	0.19
6	3416	got	0.18
7	2581	great game rip ghost	0.18
8	4731	people used play ghost game	0.18
9	3655	dont drop soap	0.17
10	5065	bought played thing saw multiplayer explosions...	0.17

Interpretação das Similaridades

Os valores de similaridade variam entre 0.17 e 0.24, indicando um nível moderado de proximidade semântica, que já era esperado, pois as reviews são curtas, escritas de forma informal e com vocabulário variado. Assim, faz com que o TF-IDF encontre poucas palavras idênticas entre os textos.

O TF-IDF atribui maior relevância às palavras que aparecem com frequência no documento. Dessa forma, as palavras como “ghost”, “soap”, “price” e “makarov” adquiriram grande relevância e tiveram um impacto significativo nas semelhanças identificadas.

As avaliações mais próximas exibem uma grande concordância nos principais termos, além de citarem personagens e eventos semelhantes do jogos, o que justifica a similaridade próxima de 0.24.

Abaixo estão os ângulos das 10 reviews com o maior cálculo da similaridade do cosseno:

Ângulos correspondentes às similaridades

Similaridade	Ângulo aproximado
0.24	76,1°
0.23	76,7°
0.20	78,5°
0.19	79,0°
0.18	79,6°
0.17	80,2°

Os resultados mostram que as reviews não são idênticas, mas compartilham informações e termos do jogo em comum, principalmente os personagens “Ghost” e “Soap”.

Conclusão

O projeto demonstrou a eficácia do uso combinado das técnicas TF-IDF e similaridade de cosseno para identificar as similaridades das avaliações de usuários ao jogo Call of Duty Modern Warfare 2 (2009).

A análise mostrou que é possível detectar termos comuns e padrões de linguagem entre diferentes review. Mesmo sendo avaliações curtas e informais.