Avaliação A3

Nomes:

Gustavo Cristhian Almeida Silva - 120112419
David Jacobis Furtado - 12011714
Leonardo Henrique Alves Silva - 12013982
Matheus Henrique Ribeiro - 120117172

Título: Sistema de Recomendação de Filmes

Introdução:

Objetivo: Este documento tem como finalidade apresentar a descrição da avaliação A3, na qual foi sugerido temas para a criação de modelos de inteligência artificial. Dentre os temas, o grupo em questão optou por fazer um sistema de recomendação de filmes, semelhante à Netflix. As recomendações de filmes baseadas em análise de tendências usando dados obtidos em uma base de filmes, gêneros de filmes, usuários e suas avaliações por filme.

Contexto: Foi solicitado ao grupo a criação de uma rede neural que fizesse recomendações de filmes baseada em perfis de usuários e no histórico de avaliações positivas de cada usuário.

Arquitetura da Rede Neural:

Tipos de camadas: Descrição das camadas da rede (entrada, ocultas, saída).

Hiperparâmetros: Lista dos parâmetros definidos (número de neurônios, taxa de aprendizado, função de ativação, etc.).

Item Based Collaborative Filtering

Item-Based Collaborative Filtering (ICF) é uma técnica utilizada em sistemas de recomendação para sugerir itens a usuários com base em suas preferências anteriores e nas preferências de usuários semelhantes. Ao contrário de métodos de filtragem colaborativa baseados em usuário, que se concentram em encontrar usuários semelhantes, o ICF se baseia na similaridade entre os itens.

Conjunto de Dados:

Origem: Dois datasets. Um contém uma ampla lista de filmes numerados por um ID e seus gêneros. O outro contém o usuários que para eles foram um conjunto de dados fictícios como nomes, sobrenomes, estado de origem, salário e etc, para que seja feita uma melhor análise de um perfil de usuário, separado por seu ID, dados os IDs dos filmes do outro dataset, e a avaliação do usuário para determinado filme.

Pré-processamento: Foram feitas análises de tendências antes do treinamento da rede neural, que serão detalhadas abaixo.
Antes das análises iniciarem foi criada uma seed que salvasse dados de perfil gerados aleatoriamente para que fosse possível associar aos dados dos datasets para fazer as análises de forma que fosse possível agrupar usuários de preferências semelhantes. Todas as etapas foram testadas previamente antes da etapa de treinamento e criação da rede neural.

Desenvolvimento:

Algoritmo de Otimização: Toda a programação foi feita usando a plataforma Google Colab e um Notebook Python para trabalhar os dados e o treinamento da máquina. Bibliotecas usadas: pandas, missingno, matplotlib.pyplot, seaborn, datetime, numpy, faker, sklearn.metrics.pairwise e surprise.

Métricas de Avaliação: À medida que um perfil é traçado de acordo com idade, renda, estado de origem e a avaliação dos usuários a máquina deve recomendar filmes que esse usuário poderia ter interesse levando em consideração tendências analisadas com os dados citados anteriormente.
Com a apresentação de dados dos filmes assistidos e rating é possível com o modelo Item Based Collaborative Filtering, recomendar filmes para o usuário utilizando de seu histórico e avaliações.

Resultados:

Desempenho: É possível considerar um desempenho satisfatório do modelo, em que as recomendações estão condizentes com cada perfil capturado.

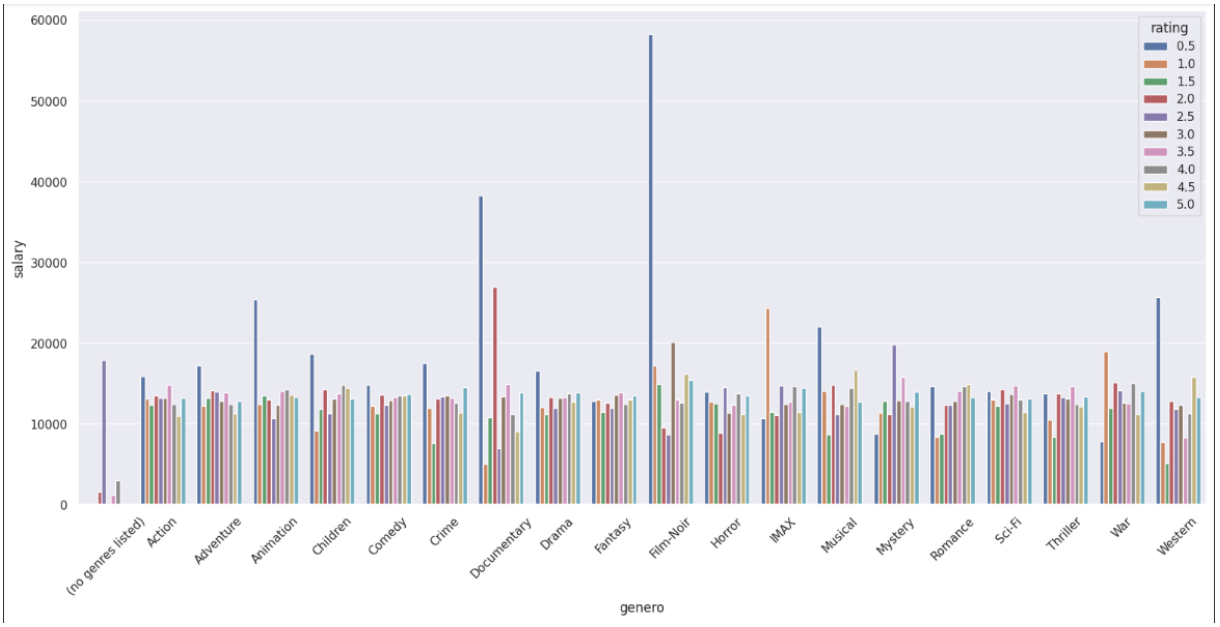
Visualizações:

Exemplo de criação de perfis de usuário:

	movieId	title	genres	userId	rating	timestamp	age	salary	full_name	states
0	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy	2	5.0	859046895	31	7431.47	Allison Ward	Rio de Janeiro
1	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy	5	4.0	1303501039	14	1200.00	Christopher Holden	Goiás
2	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy	8	5.0	858610933	56	2346.89	Keith Young	Minas Gerais
3	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy	11	4.0	850815810	12	1200.00	Joshua Smith	Pernambuco
4	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy	14	4.0	851766286	58	3556.93	Taylor Lee	Roraima

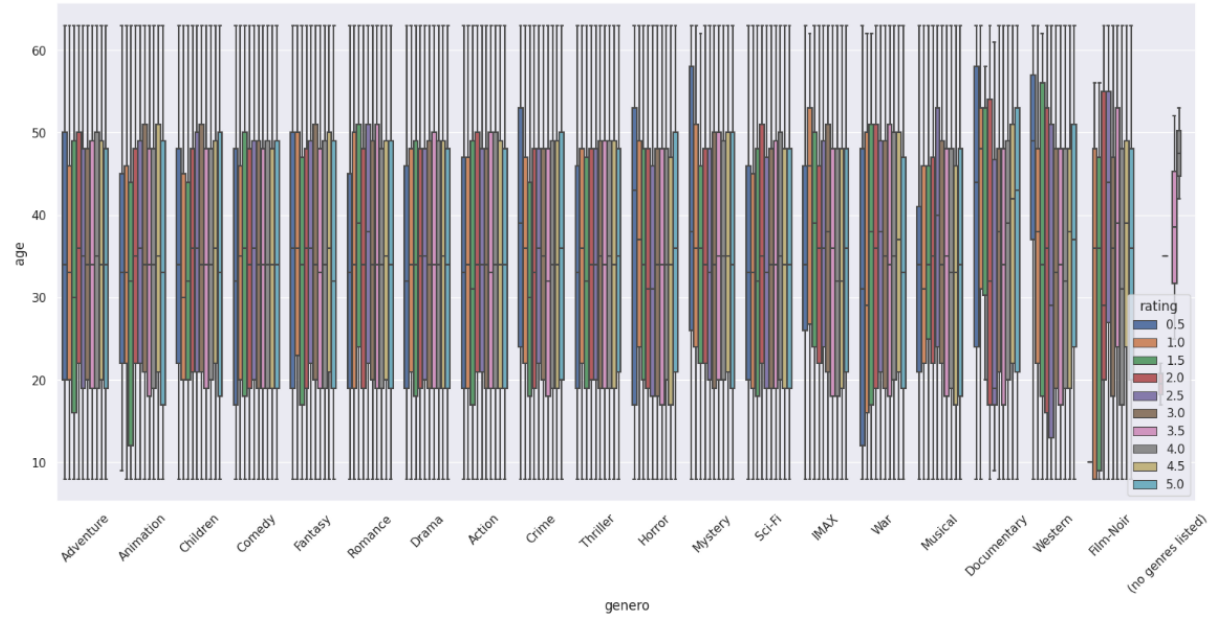
Por meio de uma seed para armazenar e de bibliotecas específicas foram gerados dados aleatórios e construídos perfis que contivessem idade, faixa salarial, estado onde mora e o nome do usuário. Ao separar as avaliações por perfis foi associado um ID de usuário a cada perfil, e sua avaliação para determinado filme. Com essa associação é possível criar tendências para as preferências de cada um.
Para a criação de nomes aleatórios foi utilizada a biblioteca Faker, que consiste em criar dados fake.
Para a criação de Gráficos foram utilizadas as seguintes bibliotecas: Seaborn,Missingno e Plotly.Express.

Gráfico 1:



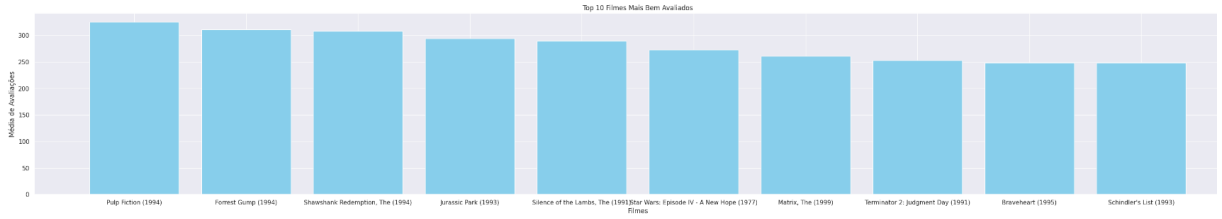
Nesse gráfico a ideia é ter uma amostra das avaliações dos filmes de acordo com a faixa salarial dos perfis. Nesse exemplo é possível ver que pessoas de maior renda tendem a gostar menos de filmes investigativos ou de documentários.É possível reparar também que notas medianas aparecem mais conforme a faixa salarial aumenta. Porém não existe uma variação considerável para considerar como base para o desenvolvimento de modelo de recomendação com base no perfil de usuário.

Gráfico 2:



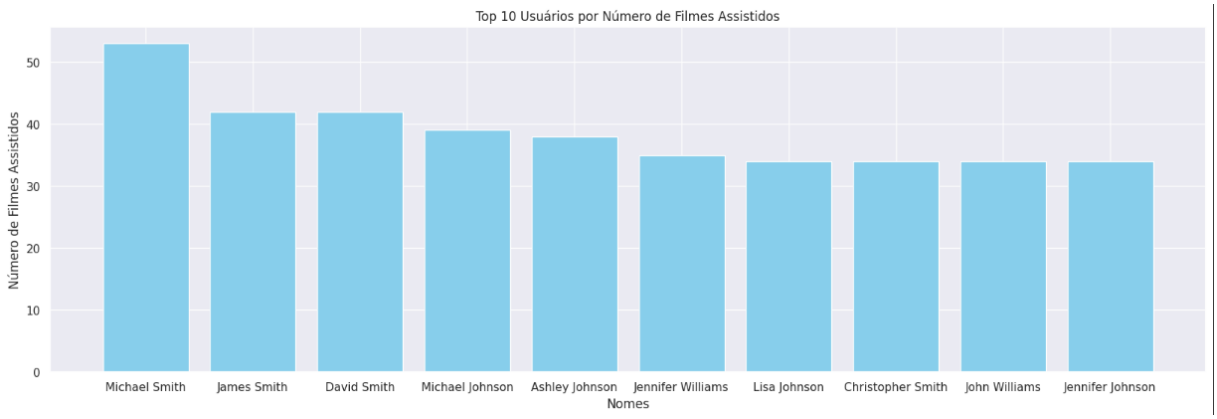
Semelhante ao gráfico anterior, a diferença é que este usa a idade como referência em vez da faixa salarial. Aqui é possível visualizar que pessoas acima dos 50 anos tendem a não assistir ou não avaliar filmes, bem como pessoas abaixo dos 10 anos, com algumas exceções. É possível também notar que a idade não é o melhor parâmetro possível se for avaliada individualmente, dada a pouca variância das avaliações entre os gêneros. Isso sugere que independente da idade, há pessoas que gostam e desgostam de todos os gêneros.

Gráfico 3:



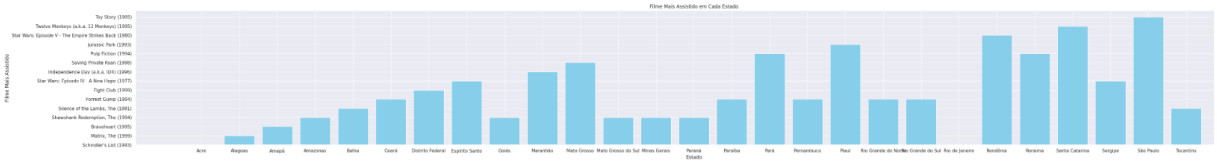
Esse gráfico coleta os 10 filmes mais bem avaliados, desta forma podendo ter uma melhor noção de análise exploratória para entender o porquê desses filmes serem os mais bem avaliados, visando no tipo de gênero do filme e avaliação.

Gráfico 4:



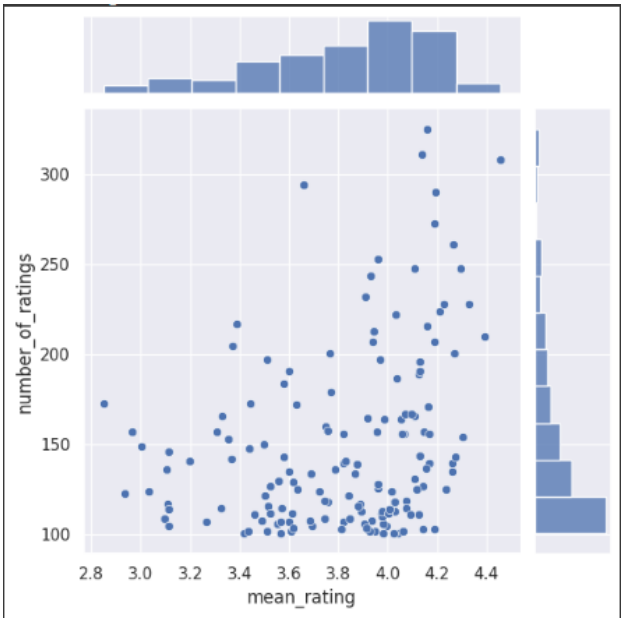
Nesse gráfico a ideia foi coletar quais são os usuários considerados mais engajados, ou seja, os 10 usuários que mais avaliam filmes. Com essa coleta é possível concluir que esses usuários também são os que mais assistem filmes. Porém não é uma análise que possa ser utilizada para a criação do modelo, pois não apresenta conteúdo que possa auxiliar no desenvolvimento do modelo.

Gráfico 5:



Nesse gráfico é possível visualizar quais foram os filmes com maior quantidade de avaliações separadas por estado brasileiro. A ideia é notar se há uma tendência em um determinado estado ter interesse maior por um gênero de filme.

Gráfico 6:



Com esse gráfico é possível ver que há uma concentração maior de avaliações consideradas médias. Dada a dispersão é possível concluir que apesar de muitos usuários terem avaliado os filmes com boas notas, a maior quantidade de avaliações tem notas próximas ao 4,0.

Exemplo de predição de filmes:

	title	rating
0	Austin Powers: The Spy Who Shagged Me (1999)	1.893162
1	Die Hard: With a Vengeance (1995)	1.500000
2	Monty Python's Life of Brian (1979)	1.189320
3	Austin Powers: International Man of Mystery (1...	1.084158
4	Die Hard (1988)	1.081818

Para iniciar a análise com a predição foi criado um algoritmo que escolhia um usuário e um filme previamente citado, ao receber tais dados o código retorna todos os filmes que o usuário escolhido assistiu, e é apresentada sua taxa de rating, em que valores altos como 1.893… representam um rating alto, já valores negativos apresentam um rate baixo, apresentando o resultado acima.

Exemplo de taxa de similaridade:

```
# Pontuação de similaridade do filme Star Wars: Episode IV - A New Hope (1977) com todos os outros filmes
picked_movie_similarity_score = item_similarity[[filme_escolhido]].reset_index().rename(columns={'Star Wars: Episode IV - A New Hope (1977)':'similarity_score'})

# Classifica as semelhanças entre os filmes avaliados pelo usuário 7 e Star Wars: Episódio IV.
n = 5
picked_userid_watched_similarity = pd.merge(left=picked_userid_watched,
                                            right=picked_movie_similarity_score,
                                            on='title',
                                            how='inner')\
                                   .sort_values('similarity_score', ascending=False)[:5]

# Dê uma olhada nos filmes assistidos pelo usuário 1 com maior similaridade
picked_userid_watched_similarity
```

	title	rating	similarity_score
20	Star Wars: Episode IV - A New Hope (1977)	0.811355	1.000000
23	Star Wars: Episode V - The Empire Strikes Back...	0.771930	0.725627
10	Star Wars: Episode VI - Return of the Jedi (1983)	0.968468	0.657868
3	Austin Powers: International Man of Mystery (1...	1.084158	0.541216
58	Pirates of the Caribbean: The Curse of the Bla...	-1.329787	0.428312

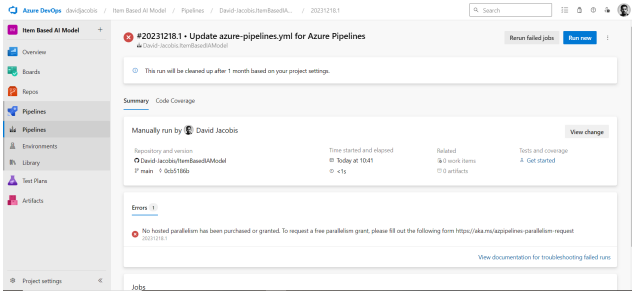
Ao escolher o filme Star Wars Episode IV - A New Hope (1977) como base, o algoritmo buscou no dataset filmes de gêneros semelhantes e listou alguns deles com base decrescente no grau de similaridade.

Recomendação baseada em histórico:

```
History-based recommendation: [('Incredibles, The (2004)', 3.935052),
('Crouching Tiger, Hidden Dragon (Wo hu cang long) (2000)', 1.435516),
('Apocalypse Now (1979)', 0.649829)]
```

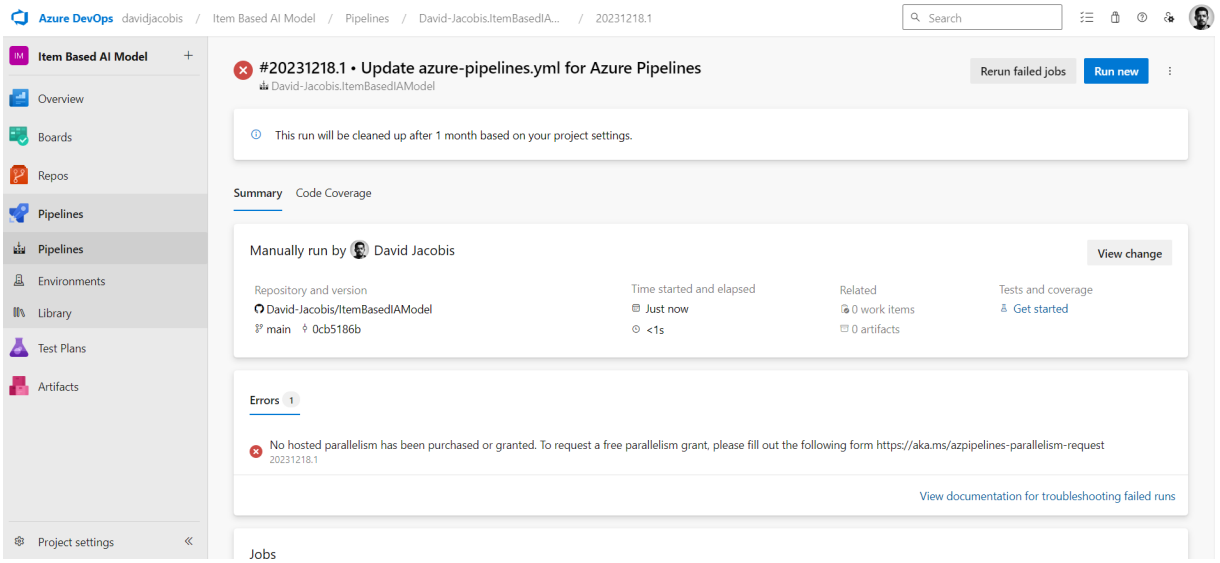
Por fim temos a função que faz a recomendação baseada no histórico, essa função utiliza o modelo Item Based para recomendação em que com base na similaridade entre os itens são realizadas previsões de recomendação dos filmes utilizando dados de filmes assistidos e não assistidos pelo usuário, e sua avaliação em ambos, para que seja realizada a média ponderada das pontuações de similaridade e as avaliações do usuário escolhido e apresentando os filmes mais recomendados com base nestes dados.

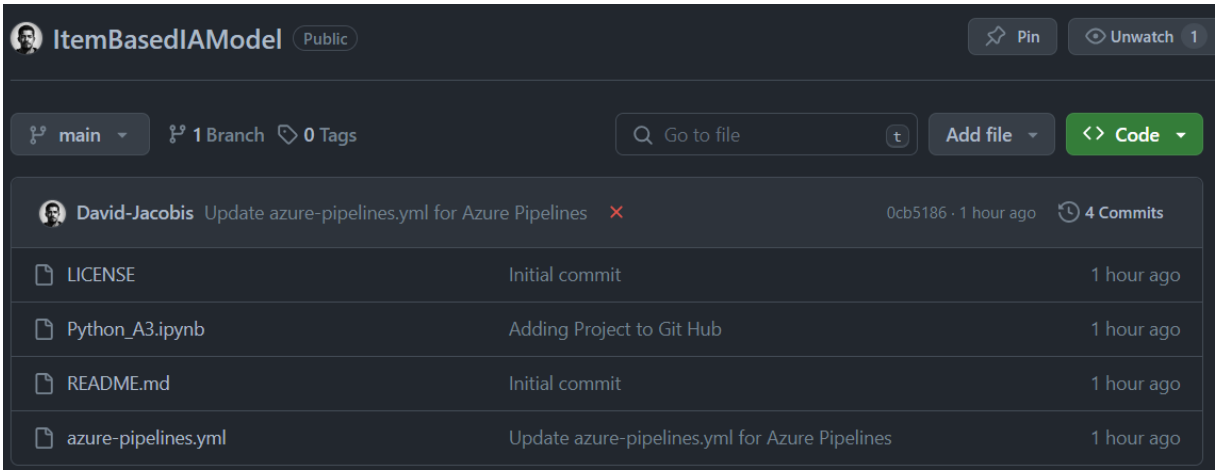
Após a criação do modelo, foi realizada a criação de um pipeline teórico de deploy completo em uma plataforma nuvem (Microsoft Azure) porém não foi possível visualizar sua execução pois era necessário pagar para o serviço escolhido.



Desafios:

Com a base de dados recebida, não tínhamos um perfil de usuário, apenas dados de filmes, rating e também gênero. Devido a estas circunstâncias criamos dados fictícios para poder ter um perfil de usuário, utilizando Bibliotecas para realizar o mesmo. Apesar dos dados criados e análises exploratórias feitas, infelizmente não tivemos uma variação de dados considerável para ser criado um modelo com base no perfil do usuário. Ao ser criada a pipeline teórica de deploy do projeto realizado, foi utilizada a plataforma de nuvem da Azure, apesar da Pipeline ser criada, não foi possível visualizar sua execução pois era necessário pagar o serviço para que ela estivesse em execução.





Conclusão:

Durante este projeto de IA, exploramos e aplicamos uma variedade de técnicas, algoritmos e métodos para resolver problemas complexos. Mergulhamos em conceitos fundamentais da IA, como aprendizado de máquina, redes neurais, algoritmos de otimização e processamento de linguagem natural. Utilizamos ferramentas e bibliotecas populares para análise e manipulação de dados, modelagem de algoritmos e avaliação de desempenho.