**Projeto de Pesquisa e Planejamento de Atividades**

|  |  |  |
| --- | --- | --- |
| **Aluno**: Gabriel Felix dos Santos | | **Data início curso**: 25/04/2023 |
| **Orientador**: Prof. Dr. Renato Máximo Sátiro | | **Defesa em:** Mês/Ano |
| **Curso**: MBA Data Science e Analytics | **Modalidade**: Distância | Turma: 231 |

**1. Título do projeto**

Do demográfico ao híbrido: performances e resultados de algoritmos de recomendação

**2. Introdução**

Em nenhum momento da história, a humanidade esteve tão interligada como na era atual, graças aos avanços computacionais e à acessibilidade a rede de internet. Relações pessoais agora estão a apenas um clique de distância e milhões de informações são trocadas a todo momento, desde textos até vídeos e áudios.

Esta abundância de dados favoreceu, e muito, a população mundial. Basta realizar uma pequena pesquisa sobre um determinado assunto que o usuário deparar-se-á com diversos conteúdos e até mesmo pesquisas científicas atuais do ramo. No entanto, também apresenta desafios significativos, pois o ser humano não consegue lidar com muitas informações simultâneas e, consequentemente, acaba tendo dificuldades de filtrar os conteúdos que o interessa Silva (2014), se desgasta e toma péssimas decisões (Ricci et al., 2011).

Devido a estes problemas, os Sistemas de Recomendação [SR] foram desenvolvidos, no qual a pesquisa ministrada por Whittaker e Sidner (1996) na classificação e filtragem de e-mails consiste em uma das primeiras aplicações da abordagem.

Com o passar do tempo, novas formas de SRs foram desenvolvidas e aperfeiçoadas e, atualmente, destacam-se alguns grupos de algoritmos, como os de Filtragem Demográfica, que agrupam os usuários em comunidades e realizam recomendações específicas para cada nicho; os de Filtragem Baseada em Conteúdo, que indicam itens similares aos que os usuários avaliaram positivamente; os de Filtragem Colaborativa, que agrupam os usuários com gostos semelhantes e realizam recomendações de itens de acordo com as similaridades entre estes grupos; e os de Filtragem Híbrida, que combinam duas ou mais abordagens de Filtragem para fornecerem melhores recomendações (Burke, 2002).

O motivo pelo qual há diferentes metodologias de filtragem de conteúdo está relacionado aos problemas encontrados nas aplicações dos SRs. Alguns algoritmos realizam recomendações muito próximas e flexíveis aos gostos dos usuários, mas em troca de uma quantidade de dados necessariamente vasta e um alto custo de processamento e tempo computacional no treinamento dos modelos de Inteligência Artificial. Outros algoritmos são menos custosos, porém fornecem recomendações ruins e generalizadas, como é o que acontece na Filtragem Baseada em Conteúdo, pois os gostos dos usuários não são sempre identificados apenas pela similaridade dos itens (Das et al., 2007).

Portanto, devido a estas problemáticas, o presente trabalho foi desenvolvido a fim de analisar o desempenho em termos de custo de hardware e tempo de execução das principais abordagens de recomendação descritas anteriormente a fim de pontuar os prós e contras de cada uma. Além de, como objetivo secundário, comparar as recomendações geradas pelos modelos.

**3. Objetivo**

O objetivo principal deste trabalho consiste em analisar o desempenho em termos de custo de hardware e tempo de execução das principais abordagens de recomendação – Filtragem Demográfica, Filtragem Baseada em Conteúdo via sinopse e metadados, Filtragem Colaborativa via similaridade de usuários e similaridade de itens, e Filtragem Híbrida –, enquanto o objetivo secundário consiste em comparar as recomendações geradas pelos modelos.

**4. Material e Métodos**

Este trabalho utiliza o método de experimentação e, a fim de facilitar a compreensão dos materiais e métodos aplicados, esta seção foi dividida em seis partes: Ambiente de Desenvolvimento, Coleta de Dados, Dicionário das Variáveis, Limpeza e Transformação dos Dados, Modelos de Recomendação, e Comparação das Performances e dos Resultados.

**Ambiente de Desenvolvimento**

Os códigos foram desenvolvidos na versão 3.10.9 da linguagem de programação Python nos ambientes de desenvolvimento Jupyter Lab e Jupyter Notebook, ambos na versão 3.5.3. Destacam-se os modelos de Filtragem Colaborativa, para os quais foi empregado o pacote Surprise, exigindo a ferramenta Microsoft C++ Built Tools versão 14 para Windows. Além do mais, pacotes do Python foram aplicados para otimizar o processo de desenvolvimento, com suas versões e finalidades listadas na Tabela 1.

Tabela 1. Pacotes da linguagem Python aplicados

|  |  |  |  |
| --- | --- | --- | --- |
| Aplicação | Pacote | Versão | Uso |
| Avaliação de performance | Psutil | 5.9.0 | Medição do uso de “Random-Access Memory [RAM]” e de processador |
| Avaliação de performance | Time | 3.10.9 | Medição do tempo de execução dos códigos |
| Modelagem e validação | LightFM | 1.17 | Treinamento e validação do modelo de Filtragem Híbrida |
| Modelagem e validação | SKLearn | 1.2.1 | Treinamento e validação dos modelos de Filtragem Baseada em Conteúdo |
| Modelagem e validação | Spacy | 3.7.2 | Preparação dos dados para os modelos de Filtragem Baseada em Conteúdo |
| Modelagem e validação | Surprise | 1.1.3 | Treinamento e Validação dos modelos de Filtragem Colaborativa |
| Modelagem e validação | WordCloud | 1.9.2 | Exploração da Bolsa de Palavras nos modelos de Filtragem Baseada em Conteúdo |
| Processamento de dados | Chardet | 4.0.0 | Checagem e transformação do formato de codificação das bases de dados para “Unicode Transformation Format 8 [UTF-8]” |
| Processamento de dados | Numpy | 1.23.5 | Manipulação das bases de dados |
| Processamento de dados | Pandas | 1.5.3 | Leitura e manipulação das bases de dados |
| Processamento de dados | Re | 2.2.1 | Aplicação de expressões regulares na limpeza e transformação dos dados |
| Visualização de dados | Matplotlib | 3.6.3 | Criação de gráficos |
| Visualização de dados | Mplcyberpunk | 0.7.0 | Estilização de gráficos |
| Visualização de dados | Seaborn | 0.12.2 | Criação de gráficos |

Fonte: Dados originais da pesquisa

Por fim, vale salientar que os códigos foram executados em um notebook com processador Intel® Core™ i5-8300H com frequência de 2,30GHz, placa de vídeo NVIDIA GeForce GTX 1050, 12 GB de RAM e sistema operacional Windows 10 baseado em 64 bits.

**Coleta de Dados**

As bases de dados encontram-se publicamente disponíveis na plataforma Kaggle e consistem em informações de animes, usuários e avaliações dos usuários sobre os animes assistidos em um dos principais sites do ramo.

A coleta foi realizada por um usuário do Kaggle utilizando a “Application Programming Interface [API]” Jikan durante 06 jul. 2023 a 06 out. 2023 abrangendo dados até o final do processo.

Foram exportadas três bases de dados em formato “Comma-Separated Values [CSV]”, no qual a primeira contém informações sobre os animes disponíveis, com 24 variáveis e 24.905 observações. A segunda contém informações sobre os usuários disponíveis, com 16 variáveis e 731.290 observações. E a terceira é constituída pelas avaliações dos usuários sobre os animes assistidos, com cinco variáveis e 24.325.191 observações.

**Dicionário das Variáveis**

O próximo passo foi realizar o estudo das bases e entender o que cada variável representa, resultando nos dicionários de variáveis abaixo representados pela Tabela 2, Tabela 3 e Tabela 4.

Tabela 2. Dicionário das variáveis da base de dados de animes

| Variável | Descrição | Tipo de dado | Amostra |
| --- | --- | --- | --- |
| Anime ID | Número de identificação | Número inteiro | 1 |
| Name | Nome original | Texto | Cowboy Bebop |
| English Name | Versão do nome em Inglês | Texto | Cowboy Bebop |
| Other Name | Versão do nome em Japonês | Texto | カウボーイビバップ |
| Score | Pontuação média seguindo a equação Bayesiana | Texto | 8,75 |
| Genres | Gêneros atribuídos | Texto | Action, Award Winning, Sci-Fi |
| Synopsis | Breve descrição da história | Texto | Crime is timeless. By the year 2071... |
| Type | Formato da animação | Texto | TV |
| Episodes | Número de episódios. Filmes são considerados como um único episódio | Texto | 26,0 |
| Aired | Período do início e final do lançamento | Texto | Apr 3, 1998 to Apr 24, 1999 |
| Premiered | Temporada de lançamento | Texto | Spring 1998 |
| Status | Status de lançamento | Texto | Finished Airing |
| Producers | Produtoras envolvidas na produção | Texto | Bandai Visual |
| Licensors | Licenciadoras e plataformas de vídeo envolvidas na distribuição | Texto | Funimation, Bandai Entertainment |
| Studios | Estúdios envolvidos na produção | Texto | Sunrise |
| Source | Material original da história | Texto | Original |
| Duration | Duração de cada episódio | Texto | 24 min per ep |
| Rating | Classificação etária | Texto | R - 17+ (violence & profanity) |
| Rank | Classificação na plataforma pelo “Score” | Texto | 41,0 |
| Popularity | Classificação na plataforma pela popularidade do momento | Número inteiro | 43 |
| Favorites | Quantidade de usuários que marcaram o item como favorito | Número inteiro | 78.525 |
| Scored By | Quantidade de usuários que avaliaram o item | Texto | 914.193,0 |
| Members | Quantidade de usuários que incluíram o item na lista para assistir | Número inteiro | 1.771.505 |
| Image Url | Imagem promocional | Texto | https://cdn.myanimelist.net/images/anime/4/19644.jpg |

Fonte: Dados originais da pesquisa

Tabela 3. Dicionário das variáveis da base de dados de usuários

| Variável | Descrição | Tipo de dado | Amostra |
| --- | --- | --- | --- |
| MAL ID | Número de identificação | Número inteiro | 1 |
| Username | Nome na plataforma | Texto | Xinil |
| Gender | Gênero sexual | Texto | Male |
| Birthday | Data de nascimento | Texto | 1985-03-04T00:00:00+00:00 |
| Location | Localização, podendo ser expressa em país, estado e cidade ou na combinação dos três | Texto | California |
| Joined | Data de quando a conta foi criada | Texto | 2004-11-05T00:00:00+00:00 |
| Days Watched | Quantidade de dias assistidos na plataforma | Número decimal | 142,3 |
| Mean Score | Pontuação média das avaliações seguindo a Média Bayesiana | Número decimal | 7,37 |
| Watching | Quantidade de itens sendo assistidos | Número decimal | 1,0 |
| Completed | Quantidade de itens já completamente assistidos | Número decimal | 233,0 |
| On Hold | Quantidade de itens que o usuário parou de assistir mas deixou na lista | Número decimal | 8,0 |
| Dropped | Quantidade de itens que o usuário parou de assistir e retirou da lista | Número decimal | 93,0 |
| Plan to Watch | Quantidade de itens que o usuário adicionou na lista e ainda não começou a assistir | Número decimal | 64,0 |
| Total Entries | Quantidade total de itens na lista | Número decimal | 399,0 |
| Rewatched | Quantidade de itens reassistidos | Número decimal | 60,0 |
| Episodes Watched | Quantidade de episódios assistidos de todos os itens | Número decimal | 8.458,0 |

Fonte: Dados originais da pesquisa

Tabela 4. Dicionário das variáveis da base de dados de avaliações

|  |  |  |  |
| --- | --- | --- | --- |
| Variável | Descrição | Tipo de dado | Amostra |
| User ID | Número de identificação do usuário na plataforma | Número inteiro | 1 |
| Username | Nome do usuário na plataforma | Texto | Xinil |
| Anime ID | Número de identificação do anime na plataforma | Número inteiro | 21 |
| Anime Title | Nome original do anime na plataforma | Texto | One Piece |
| Rating | Pontuação que o usuário atribuiu ao anime | Número inteiro | 9 |

Fonte: Dados originais da pesquisa

**Limpeza e Transformação dos Dados**

Esta etapa foi dividida em quatro partes, sendo a primeira composta por limpezas e transformações aplicadas nas três bases de dados de forma geral, enquanto as demais consistem em processos aplicados individualmente em cada uma.

**Limpezas e Transformações Gerais**

Todas as bases de dados passaram pela checagem e transformação da codificação do arquivo para o formato UTF-8, além de terem as variáveis padronizadas em minúsculas, uso de sublinhados no lugar de espaços e palavras no idioma Inglês Americano. Os nomes finais das variáveis são os abaixo:

• Base de Dados de Animes: “id”, “title”, “english\_title”, “japanese\_title”, “score”, “genres”, “synopsis”, “type”, “episodes”, “aired”, “premiered”, “status”, “producers”, “licensors”, “studios”, “source”, “duration”, “rating”, “rank”, “popularity”, “favorites”, “scored\_by”, “members” e “image\_url”;

• Base de Dados de Usuários: “id”, “name”, “gender”, “birthday”, “location”, “joined”, “days\_watched”, “mean\_score”, “watching”, “completed”, “on\_hold”, “dropped”, “plan\_to\_watch”, “total\_entries”, “rewatched” e “episodes\_watched”;

• Base de Dados de Avaliações: “user\_id”, “username”, “anime\_id”, “anime\_title” e “rating”.

**Limpeza e Transformação na Base de Dados de Animes**

Foram removidos animes não anunciados ou lançados ainda, identificados pelos valores “Not yet aired” na variável “status” e “Not available” na variável “aired” respectivamente. Em seguida, foram descartadas as variáveis “english\_title”, “japanese\_title”, “aired” e “premiered”, pois somente a variável “title” foi utilizada para identificar os itens pelo nome, e os itens não anunciados e/ou lançados já foram removidos.

Depois, foi necessário tratar os valores ausentes, substituindo todos os “UNKNOWN” por -1 nas variáveis “score”, “episodes”, “rank” e “scored\_by” e converter o tipo destas variáveis de texto para número inteiro; enquanto que os “UNKNOWN” em “genres”, “producers”, “licensors”, “studios” e “’rating” foram substituídos por um hífen “-”. Já na variável “synopsis”, os valores ausentes estavam representados como “No description available for this anime” e foram também substituídos por um hífen “-”.

Após filtrar as observações e substituir os valores ausentes, foi realizada a padronização das variáveis “title” e “synopsis”, convertendo os valores para minúsculo, sem acentos, e consistidos em apenas letras do alfabeto arábico, números, vírgulas, espaços e hifens, sendo "◯" o único caractere japonês mantido e substituído pela sua forma em "Rōmaji", "Maru". Ele foi mantido pelo fato de que há uma obra no qual o título original é apenas o caractere e, consequentemente, ficaria com o nome em branco após o descarte de todos os caracteres japoneses.

Por fim, foi necessário verificar a existência de observações duplicadas. Alguns itens possuíam o mesmo título e, após checá-los na plataforma, foi constatado de que consistem em sequências das obras, portanto, foi apenas adicionado uma numeração ao final do título. Por exemplo, o título “5-toubun no hanayom” aparece duas vezes porque o anime possui duas temporadas, logo, uma observação ficou com o título original, enquanto que a outra ficou com o título “5-toubun no hanayom 2”.

Já sobre os gêneros, produtoras, licenciadoras e estúdios, os valores duplicados foram excluídos. Tomando os gêneros como exemplo, o valor “mistery, sci-fi, mistery, action” tornar-se-ou “mistery, sci-fi, action”.

**Limpeza e Transformação na Base de Dados de Usuários**

Foram removidos usuários que desativaram as contas, identificados pelos valores nulos nas variáveis “days\_watched”, “mean\_score”, “watching”, “completed”, “on\_hold”, “dropped”, “plan\_to\_watch”, “total\_entries”, “rewatched” e “episodes\_watched”. Em seguida, foram descartados “birthday” e “episodes\_watched”, pois não seriam utilizados no estudo.

Depois, os valores ausentes foram tratados, sendo substituídos por hifens “-” em “gender”, além de forçar o nome de usuário “None” para ser do tipo de dado texto, pois este valor é considerado como nulo dependendo da versão do Python.

Após isto, todas as variáveis numéricas foram convertidas de ponto flutuante para inteiro, pois todos os valores contidos nelas são e podem ser apenas inteiros.

Posteriormente, foi verificado se todos os usuários possuem pontuação média de avaliação entre zero e dez, sendo constatado apenas um com valor acima do esperado e, portanto, sendo truncado para dez.

Por fim, os valores das variáveis “name”, “joined” e “gender” foram padronizados para minúsculo.

**Limpeza e Transformação na Base de Dados de Avaliações**

Foram removidas observações que contêm os animes e/ou os usuários que foram removidos nas duas bases de dados anteriores, além de forçar o nome de usuário “None” para ser do tipo de dado texto.

Em sequência, foi necessário realizar a padronização da variável “anime\_title” da mesma forma que foi feito nas “title” e “synopsis” da base de animes, além de tratar os títulos duplicados conforme realizado na mesma base de dados. Por fim, os valores da variável “username” foram padronizados para minúsculo.

A Tabela 5 indica a quantidade de variáveis e observações antes e depois da limpeza e transformação dos dados em cada base.

Tabela 5. Quantidade de variáveis e observações antes e depois da Limpeza e Transformação dos dados

|  | Base de dados de animes | Base de dados de usuários | Base de dados de avaliações |
| --- | --- | --- | --- |
| Variáveis iniciais | 24 | 16 | 5 |
| Variáveis finais | 21 | 14 | 5 |
| Variáveis removidas (%) | 12,5000% | 12,5000% | 0,0000% |
| Observações iniciais | 24.905 | 731.290 | 24.325.191 |
| Observações finais | 23.748 | 731.282 | 23.796.586 |
| Observações removidas (%) | 4,6456% | 0,0010% | 2,1731% |

Fonte: Dados originais da pesquisa

**Modelos de Recomendação**

Sete modelos de recomendação foram criados neste trabalho, sendo dois de Filtragem Demográfica, dois de Filtragem Baseada em Conteúdo, dois de Filtragem Colaborativa e um de Filtragem Híbrida. As recomendações de todos retornam o código de identificação do item, título, pontuação média, gêneros, imagem promocional e pontuação estimada dos animes.

Além do mais, apenas as bases de dados de usuários e avaliações foram utilizadas no processo de treinamento, validação e recomendação, sendo a primeira base aplicada em todos os modelos, e a segunda, somente nos de Filtragem Colaborativa e Filtragem Híbrida.

O processo de preparação dos dados, treinamento e validação de cada modelo foram descritos nas subseções Filtragem Demográfica, Filtragem Baseada em Conteúdo, Filtragem Colaborativa e Filtragem Híbrida.

**Filtragem Demográfica**

Dois modelos foram criados utilizando a metodologia de Filtragem Demográfica, enquanto o primeiro (modelo A) realiza recomendações dos itens com maiores pontuações de Média Bayesiana, o segundo (modelo B) realiza recomendações levando em consideração a popularidade dos itens entre os usuários.

Como a plataforma onde os dados foram coletados utiliza a Média Bayesiana para as pontuações dos animes[[1]](#footnote-2), o modelo A precisou apenas ordenar os itens pela variável “score” de modo descendente e retornar os dez com maiores valores.

Já o modelo B trabalhou com a variável “popularity”, ordenando os itens de modo ascendente e retornando os dez com menores valores, ou seja, os dez mais acima na classificação de popularidade.

Caso fosse necessário realizar o cálculo da Média Bayesiana no primeiro modelo, aplicar-se-ia a fórmula (1) (Khatri, 2017).

|  | (1) |
| --- | --- |

onde c: é a quantidade mínima de avaliações que o item necessita obter para ser incluído na Média Bayesiana; m: é a Média Aritmética de toda a base de dados; n: é a quantidade de avaliações que o item obteve; e r: é a Média Aritmética do item.

**Filtragem Baseada em Conteúdo**

Dois modelos foram criados utilizando a metodologia de Filtragem Baseada em Conteúdo, enquanto o primeiro (modelo C) realiza recomendações de itens com sinopses similares, o segundo (modelo D) realiza recomendações de itens com os metadados de gêneros, tipos e fontes originais semelhantes. Todos os métodos abaixo, com exceção da remoção de palavras e da lematização, foram aplicados em ambos os modelos, sendo as variáveis utilizadas a única diferença no processo.

Inicialmente, uma nuvem de palavras foi criada para verificar os termos mais frequentes nas variáveis. Após isto, todas as palavras foram padronizadas em minúsculas, sem quebras de linha, sem caracteres especiais e sem pontuações.

Em seguida e apenas no modelo C, todas as Palavras de Parada foram removidas, pois seus significados não tinham tanta relevância e iriam apenas enviesar o treinamento do modelo. Também, todos os pronomes próprios foram removidos porque foi constatado de que este tipo de palavra possuía um grande peso no cálculo de similaridade, resultando em recomendações de animes com sinopses distantes em similaridade, mas com o pronome próprio em comum.

Considerando uma palavra equivalente a um token, uma Bolsa de Palavras foi criada no qual cada elemento foi transformado em sua forma de dicionário por meio da Lematização e teve seu peso de frequência obtido aplicando a técnica Frequência do Termo – Frequência Inversa dos Documentos [FT–FID].

Depois, a similaridade entre cada elemento foi calculada utilizando a Similaridade do Cosseno, dada pela equação (2) (Rahutomo et al., 2012).

|  | (2) |
| --- | --- |

onde Wq e Wd: são as observações utilizadas no treinamento do modelo; k: é cada palavra presente na Bolsa de Palavras; t: é a quantidade de palavras na Bolsa de Palavras; e Wqk e Wdk: são a frequência da palavra “k” nas observações “Wq” e “Wd”.

Por fim, foi criada a função de recomendação de itens, que recebe o título do item como parâmetro e retorna os dez animes mais similares em questão de sinopse (modelo C) e de metadados (modelo D).

**Filtragem Colaborativa**

Dois modelos foram criados utilizando a metodologia de Filtragem Colaborativa, enquanto o primeiro (modelo E) realiza recomendações de itens que usuários semelhantes avaliaram positivamente, o segundo (modelo F) realiza recomendações de itens semelhantes aos que o usuário selecionado avaliou positivamente, sendo todos os itens recomendados consistidos em animes que usuários semelhantes ao selecionado avaliaram positivamente.

Iniciando pelo modelo E, as variáveis “username” e “anime\_title” foram descartadas e os melhores valores para os parâmetros de ajustes foram encontrados utilizando a técnica de Ajuste de Parâmetros com Pesquisa Aleatória e Validação Cruzada com a base de dados reduzida significativamente de 23 milhões de amostras para 15 mil. A redução foi necessária devido ao fato da máquina do autor não suportar o processo com uma quantidade maior de amostras, tornando-se lenta e chegando até a travar.

Depois a base de dados foi separada em treino e validação, além da variável “rating” ser convertida da escala de zero a dez para zero a cinco, pois o pacote Surprise, utilizado no treinamento e validação do modelo, trabalha apenas nesta escala.

Posteriormente, foi realizado o treinamento do modelo com a técnica K-Vizinhos Mais Próximos com Médias e a validação com a Validação Cruzada. Os cálculos das distâncias dos pontos foram feitos aplicando a fórmula da Distância Euclidiana (3) (Chomboom et al., 2015).

|  | (3) |
| --- | --- |

onde xs e yt: são as coordenadas dos pontos no eixo horizontal; e xs’ e yt’: são as coordenadas dos pontos no eixo vertical.

Por fim, foi criada a função de recomendação, além de converter a pontuação estimada da escala de zero a cinco para zero a dez.

O modelo F também passou pelos mesmos processos, sendo o aumento da base de dados para 250 mil amostras e o uso da Decomposição de Valores Singulares tanto no Ajuste de Parâmetros quanto na validação do modelo as únicas diferenças.

**Filtragem Híbrida**

Apenas um modelo (modelo G) foi criado utilizando a metodologia de Filtragem Híbrida. Nele, as variáveis “username” e “anime\_title” foram descartadas e a base de dados foi reduzida significativamente para 250 mil amostras pelo mesmo motivo descrito na seção de Filtragem Colaborativa.

Do mesmo modo do pacote “Surprise”, o “LightFM” trabalha apenas com pontuações na escala de zero a cinco, portanto, foi necessário transformar os valores da variável “rating”.

Posteriormente, a base de dados foi convertida no formato específico aceito pelo pacote, além de ser montada a matriz de interação dos usuários com cada anime. Depois o modelo foi treinado com o algoritmo “LightFM” e validado com a Curva Característica de Operação do Receptor [COR], mais conhecida como “Receiving Operation Characteristic [ROC] Curve” do Inglês Americano.

Por fim, foi criada a função de recomendação, além de converter a pontuação estimada da escala de zero a cinco para a de zero a dez.

**Comparação das Performances e dos Resultados**

As comparações não foram realizadas pelo autor no presente momento de escrita deste trabalho, mas os métodos já foram decididos.

As performances de cada modelo serão comparadas levando em consideração o uso do processador e da memória RAM da máquina durante todo o processo de preparação dos dados, treinamento e validação dos modelos, e realização de uma recomendação final; além de contabilizar o tempo gasto na execução do código em segundos.

Por outro lado, a comparação dos resultados dar-se-á pela análise das recomendações e pontos positivos e negativos de cada metologia de filtragem, bem como de cada modelo individual.

**5. Resultados Esperados**

Espera-se que os modelos de Filtragem Demográfica apresentem os menores custos de hardware e tempo de execução devido à somente aplicação da Média Bayesiana e da ordenação de popularidade para calcular as pontuações dos animes. No entanto, prevê-se que estes modelos gerem as recomendações menos precisas entre os demais, pois tenderão a recomendar os mesmos produtos a todos os usuários, ignorando os gostos individuais destes e as similaridades daqueles.

Em relação aos modelos C e D de Filtragem Baseada em Conteúdo, estima-se um aumento no custo tanto de hardware quanto de tempo em comparação com os modelos de Filtragem Demográfica. Esse aumento será devido à consideração da similaridade dos itens, seja por meio da sinopse (modelo C), seja por meio dos metadados de gênero, tipo e fonte original (modelo D). Contudo, espera-se que as recomendações sejam mais precisas, embora fiquem limitadas a animes do mesmo universo ou com temáticas semelhantes, respectivamente.

Sobre os modelos E e F de Filtragem Colaborativa, prevê-se um custo alto, talvez exigindo uma redução significativa na base de dados para tornar o treinamento do modelo viável na máquina do autor. Isso pode resultar em recomendações mais precisas e baseadas nos gostos individuais dos usuários, mas sem atingir todo o potencial dos modelos devido à redução da base de dados.

Quanto ao modelo de Filtragem Híbrida, estima-se que apresente um problema semelhante de custo, porém com uma menor redução na base de dados aplicando a técnica de Decomposição em Valores Singulares. Com isso, os resultados tendem a ser melhores do que os da Filtragem Colaborativa, mas também pode não ter o potencial total atingido.

Por fim, vale salientar algumas limitações das bases de dados neste estudo:

• Como os dados foram coletados apenas de uma plataforma, todas as recomendações resultantes são válidas apenas para esta plataforma em específico;

• Como os dados abrangem até o período de 06 out. 2023, todos os animes lançados, as contas de usuários criadas e as avaliações realizadas posteriormente não têm influência alguma nos modelos;

• Como os dados são apenas de animes, os modelos ficaram limitados a realizar recomendações apenas deste tipo de conteúdo.

**6. Cronograma de Atividades**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Atividades planejadas** | **Mês** | | | | | | | | | |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| Iniciar conversa sobre o TCC |  | **x** |  |  |  |  |  |  |  |  |
| Fechar o tema e o objetivo |  |  | **x** |  |  |  |  |  |  |  |
| Escrever a Metodologia Pretendida |  |  | **x** |  |  |  |  |  |  |  |
| Entregar o projeto ao orientador |  |  | **x** |  |  |  |  |  |  |  |
| Entregar o projeto ao MBA USP/ESALQ |  |  |  | **x** |  |  |  |  |  |  |
| Aplicar a metodologia/coletar e estruturar os dados e iniciar a discussão dos resultados |  |  |  |  |  | **x** |  |  |  |  |
| Fechar introdução, material e métodos, iniciar os resultados e discussões e enviar ao orientador |  |  |  |  |  | **x** |  |  |  |  |
| Entregar os Resultados Preliminares ao MBA USP/ESALQ |  |  |  |  |  |  | **x** |  |  |  |
| Fechar resultados e iniciar a conclusão |  |  |  |  |  |  |  |  | **x** |  |
| Fechar a conclusão e iniciar o resumo |  |  |  |  |  |  |  |  | **x** |  |
| Fechar o trabalho como um todo e enviar ao orientador |  |  |  |  |  |  |  |  | **x** |  |
| Fechar o trabalho como um todo e enviar ao MBA USP/ESALQ (entrega do TCC e agendamento da defesa) |  |  |  |  |  |  |  |  |  | **x** |
| Entrega da versão final revisada |  |  |  |  |  |  |  |  |  | **x** |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |

**Projeto de Pesquisa; Resultados Preliminares; Entrega do Trabalho de Conclusão de Curso; Entrega da Apresentação da Defesa**

**7. Referências Bibliográficas**

Burke, R. 2002. Hybrid recommender systems: Survey and experiments. Journal User Modeling and User-Adapted Interaction 12: 331-370. Disponível em: <<https://fcd.ugr.es/sites/centros/fcd/public/ficheros/PROYECTO2023/Burke2002_Article_HybridRecommenderSystemsSurvey.pdf>>. Acesso em: 20 mar. 2024.

Chomboon, K.; Chujai, P.; Teerarassamee, P.; Kerdprasop, K.; Kerdprasop, N. 2015. An Empirical Study of Distance Metrics for k-Nearest Neighbor Algorithm. In: Institute of Industrial Applications Engineers, 2015, Kitakyushu, Fukuoka, Japão. Anais... p. 280-285. Disponível em: <<https://pdfs.semanticscholar.org/815f/22363962afda432435cdbb478857547a17b5.pdf>>. Acesso em: 15 mar. 2024.

Das, A.; Datar, M.; Garg, A.; Rajaram, S. 2007. Google News Personalization: Scalable Online Collaborative Filtering. In: WWW 2007 / Track: Industrial Practice and Experience, 2007, Banff, Alberta, Canadá. Anais... p. 271-280. Disponível em: <<https://www.ri.cmu.edu/pub_files/2009/0/p271-das.pdf>>. Acesso em: 20 mar. 2024.

Khatri, A. 2017. Popularity based product rating system using Bayesian model. Dissertaçãoo de Mestrado em Engenharia da Computação. University of Victoria, Victoria, Colúmbia Britânica, Canadá. Disponível em: <https://dspace.library.uvic.ca/server/api/core/bitstreams/c615f973-11b4-417c-b196-7b4d94bd55b8/content>. Acesso em: 15 mar. 2024.

Rahutomo, F.; Kitasuka, T.; Aritsugi, M. 2012. Semantic Cosine Similarity. In: The 7th International Student Conference on Advanced Science and Technology ICAST, 2012, Seul, Coreia do Sul. Anais... p. 1-2. Disponível em: <<https://www.researchgate.net/profile/Faisal-Rahutomo/publication/262525676_Semantic_Cosine_Similarity/links/0a85e537ee3b675c1e000000/Semantic-Cosine-Similarity.pdf>>. Acesso em: 15 mar. 2024.

Ricci, F.; Rokach, L.; Shapira, B.; Kantor, P.B.; 2011. Recommender Systems Handbook. 1ed. Springer, New York, NY, USA. Disponível em: <<https://d1wqtxts1xzle7.cloudfront.net/32978074/Recommender_systems_handbook.pdf>>. Acesso em: 20 mar. 2024.

Silva, R.G.N. e. 2014. Sistema de Recomendação baseado em conteúdo textual: avaliação e comparação. Dissertação de Mestrado Multi-institucional em Ciência da Computação MMCC. Universidade Federal da Bahia e Universidade Estadual de Feira de Santana, Salvador, Bahia, Brasil. Disponível em: <<https://repositorio.ufba.br/bitstream/ri/19281/1/dissertacao_mestrado_ciencia_computacao_rafael_glauber.pdf>>. Acesso em: 20 mar. 2024.

Whittaker, S.; Sidner, C. 1996. Email overload: exploring personal information management of email. In: CHI – Conference on Human Factors in Computing Systems, 1996, Vancouver, British Columbia, Canadá. Anais... p. 276-283. Disponível em: <<https://dl.acm.org/doi/pdf/10.1145/238386.238530>>. Acesso em: 20 mar. 2024.

1. https://myanimelist.net/info.php?go=topanime [↑](#footnote-ref-2)