ANDREI SOUZA DE OLIVEIRA – RA: 10408496

DANIELE DOS SANTOS ROSA – RA: 10407781

GABRIELA OHASHI DE SOUZA – RA: 10110022

MARINA OHASHI DE SOUZA – RA: 10161483

MIGUEL MAURÍCIO TADEU PITALI DA SILVA – RA: 10407541

**PROJETO APLICADO III: BOA LEITURA**

São Paulo

2024

**RESUMO**

Os sistemas de recomendação desempenham um papel crucial na era da informação, ajudando os usuários a filtrar e descobrir conteúdo relevante em meio a uma vasta quantidade de informações disponíveis. Diante dos desafios de escalabilidade e aprimoramento das recomendações, propostas como recomendações baseadas em competências e análises de comportamento dos usuários ganham destaque. Este trabalho apresenta um projeto de recomendação de livros que busca aprimorar a precisão das recomendações e a adaptação às preferências individuais dos usuários. Para isso, utilizou-se a base de dados "Book-Crossing: User review ratings" do Kaggle e técnicas de aprendizado de máquina, incluindo o algoritmo K-Nearest Neighbors (KNN).

O KNN é um método não paramétrico que se baseia na proximidade dos exemplos de treinamento para tomar decisões de classificação ou regressão, sendo uma escolha popular para sistemas de recomendação devido à sua simplicidade e eficácia. Além disso, foram utilizadas diversas bibliotecas Python, como Matplotlib, Numpy, Pandas e Scikit-learn, para análise de dados e criação de modelos de machine learning. Este projeto visa contribuir não apenas para a filtragem e descoberta de conteúdo relevante, mas também para o avanço das técnicas de recomendação de livros, tornando-as cada vez mais precisas e personalizadas para os usuários.

**SUMÁRIO**

[INTRODUÇÃO………………………………………………………………………………. 4](#_heading=h.30j0zll)

[OBJETIVOS E METAS](#_heading=h.1fob9te).................................................................................................5

[FUNDAMENTAÇÃO TEÓRICA](#_heading=h.d9y4151796yf)…………………………………………………………….6

[METODOLOGIA](#_heading=h.vt4dlc1pwirn)…………………………………………………………………………....10

ANÁLISE EXPLORATÓRIA………………………………………………………............11 PRÉ-PROCESSAMENTO E LIMPEZA DOS DADOS………………………………….15 SISTEMA DE RECOMENDAÇÃO DE LIVROS……........................……………........19

AVALIAÇÃO DE DESEMPENHO........................…………………........…………...…24

CONCLUSÃO E TRABALHOS FUTUROS…………………………………….............26

[BIBLIOGRAFIA……………………………………………………………………………..](#_heading=h.3dy6vkm)28

**INTRODUÇÃO**

Os sistemas de recomendação são ferramentas cruciais nos dias atuais, especialmente diante da abundância de informações disponíveis. Eles desempenham um papel fundamental ao ajudar os usuários a filtrar e descobrir conteúdos relevantes em meio a um mar de opções.

Esses sistemas enfrentam desafios de escalabilidade e aprimoramento das recomendações, mas são essenciais para facilitar a tomada de decisões dos usuários e melhorar sua experiência de consumo (Medeiros, 2013).

Nesse contexto, propostas de melhoria, como recomendações baseadas em competências e análises de comportamento dos usuários, ganham destaque. A constante evolução desses sistemas é crucial para acompanhar as necessidades dinâmicas dos usuários e garantir recomendações cada vez mais precisas e personalizadas.

O presente trabalho apresenta um projeto de recomendação de livros. Além de contribuir para a filtragem e descoberta de conteúdo relevante, a pesquisa busca aprimorar a precisão das recomendações e a adaptação às preferências individuais dos usuários, fortalecendo ainda mais a relevância desses sistemas na era da informação.

**Conjunto de dados escolhido**

“Book-Crossing: User review ratings”, disponível em: [Book-Crossing: User review ratings (kaggle.com)](https://www.kaggle.com/datasets/ruchi798/bookcrossing-dataset/data)

**Repositório Projeto:**

[OhashiMarina/Projeto-Aplicado-III (github.com)](https://github.com/OhashiMarina/Projeto-Aplicado-III)

**OBJETIVOS E METAS**

Com os dados obtidos sobre os usuários, títulos de livros e avaliações dos usuários as obras, com o intuito de otimizar e melhorar a experiência do cliente Amazon, criamos um sistema de recomendação através de algoritmos de aprendizagem de máquina.

O propósito deste projeto é sugerir livros de forma colaborativa e examinar as características dos diferentes grupos de usuários em suas leituras. Os objetivos específicos deste trabalho incluem:

* Criar um sistema de recomendação de livros.
* Analisar os grupos de usuários e descrever suas relações, destacando seus padrões de comportamento.

**FUNDAMENTAÇÃO TEÓRICA**

A Análise Exploratória de Dados (EDA, *Exploratory Data Analysis*) é usada para analisar e investigar conjuntos de dados e resumir suas características principais, empregando métodos quantitativos e de visualização dos dados. Dentro do modelo CRISP-DM, a EDA compreende as fases de Entendimento e Preparação dos Dados, incluindo também a fase de Modelagem.

O modelo CRISP-DM (*Cross-Industry Standard Process for Data Mining*) é uma metodologia abrangente de mineração de dados e um modelo de processo que oferece modelo para a execução de um projeto de mineração de dados. Ele é estruturado em seis etapas distintas: compreensão do negócio, compreensão dos dados, preparação dos dados, modelagem, avaliação e implantação (Shearer, 2000).

Diagrama

Descrição gerada automaticamente

**Figura 1 - Fases do CRISP-DM (Shearer, 2000).**

A sequência de fases no CRISP-DM não segue uma ordem rígida. Na maioria dos projetos, há um movimento flexível entre as etapas, permitindo retornos quando necessário. Esta metodologia abrange descrições das fases típicas de um projeto, as tarefas necessárias em cada fase e explicações das relações entre essas tarefas. Como modelo de processo, o CRISP-DM oferece uma visão geral do ciclo de vida da mineração de dados (Chapman, 2000).

A fase inicial envolve a aquisição do conhecimento do domínio de negócios, ou seja, compreender os objetivos do projeto de mineração de dados a partir da perspectiva do negócio. Esse entendimento se transforma em um problema de mineração de dados. É amplamente reconhecido que essa etapa é uma das mais cruciais do processo, pois é a base para o desenvolvimento de um plano preliminar do projeto de mineração, direcionado para a realização dos objetivos (Shearer, 2000).

A segunda fase, que envolve a compreensão dos dados, inicia-se com a coleta inicial de dados, com o propósito de desenvolver uma familiaridade com os dados, identificar possíveis problemas de qualidade dos dados, obter *insights* iniciais e identificar subconjuntos de interesse que possam levar à formulação de hipóteses sobre informações ocultas. Esse estágio é de fundamental importância para prevenir surpresas indesejadas durante a fase subsequente, a preparação de dados, que frequentemente é a etapa mais extensa de um projeto (Shearer, 2000).

Na fase subsequente, ocorre a preparação dos dados, que engloba todas as atividades necessárias para construir o conjunto de dados final a partir dos dados brutos iniciais. Esses dados preparados servirão como entrada para a ferramenta de modelagem na etapa seguinte. As tarefas de preparação de dados são flexíveis e podem ser realizadas em várias iterações, sem uma ordem estritamente definida. Essas atividades envolvem a seleção de tabelas, registros e atributos, bem como a realização de transformações e a limpeza dos dados (Chapman et al., 2000).

A etapa de preparação de dados é a mais crítica do processo e frequentemente a que demanda maior tempo em projetos de mineração de dados. Estima-se que, em geral, essa fase absorva entre 50-70% do tempo e dos recursos de um projeto. Alocar recursos adequados para as fases iniciais de compreensão do negócio e esforços de tratamento de dados pode ajudar a minimizar a carga relacionada a essa etapa, mas, ainda assim, será necessário um esforço substancial para a preparação e formatação dos dados para fins de mineração (IBM, 2016).

A etapa de modelagem ocorre na quarta fase do processo. Dependendo da natureza do problema de mineração, diversas técnicas podem ser aplicadas. Tipicamente, a modelagem envolve várias iterações, nas quais o analista de dados executa múltiplos modelos, inicialmente com as configurações padrão e, em seguida, ajustam os parâmetros para obter valores otimizados. Além disso, é comum retornar à fase de preparação de dados, se necessário, para realizar manipulações específicas exigidas pelos modelos (Shearer, 2010; IBM, 2016).

A sexta e última fase é a etapa de implantação, na qual os novos *insights* e conhecimentos descobertos são aplicados para promover melhorias na organização. Durante essa etapa, é fundamental que todo o conhecimento adquirido seja organizado e apresentado de maneira que o cliente possa utilizá-lo eficazmente no processo de tomada de decisão. (Shearer, 2000; Chapman et al., 2000).

**APRENDIZADO DE MÁQUINA**

Nos últimos anos, houve um notável crescimento na pesquisa em aprendizado de máquina. O aprendizado de máquina é uma disciplina da inteligência artificial que se concentra no desenvolvimento de técnicas computacionais para a aquisição automática de conhecimento e na construção de sistemas capazes de aprender com base em experiências adquiridas por meio da resolução bem-sucedida de problemas anteriores (Rezende, 2005). Essa área representa a interseção entre estatística, inteligência artificial e ciência da computação, e tem aplicação significativa no reconhecimento de padrões (Guido, 2016).

O Aprendizado de Máquina (*Machine Learning*) é uma disciplina que emprega uma ampla gama de procedimentos e algoritmos para a identificação automatizada de padrões, agrupamentos e tendências nos dados, com o propósito de extrair informações valiosas para análise. Em termos simples, pode ser descrito como o uso de métodos matemáticos para treinar algoritmos a fim de reconhecer padrões (Nelli, 2015).

**APRENDIZADO DE MÁQUINA SUPERVISIONADO**

O aprendizado supervisionado é um processo que envolve a extração de um modelo de conhecimento a partir de dados apresentados na forma de pares ordenados, consistindo em uma entrada e uma saída desejada. A entrada representa o conjunto de atributos ou características que são fornecidos ao algoritmo para um caso específico, enquanto a saída desejada corresponde ao valor de uma característica-alvo que se espera que o algoritmo possa prever sempre que receber determinados valores de entrada (Goldschmidt, 2015). Alguns exemplos de algoritmos que se encaixam nesse modelo incluem K-Nearest Neighbors (KNN), Modelos Lineares, Classificador Naive Bayes, Support Vector Machines e Redes Neurais Artificiais.

No presente trabalho, optou-se pela utilização do modelo KNN, como será descrito posteriormente.

## **MÉTRICAS DE AVALIAÇÃO DE EFICIÊNCIA**

A avaliação das métricas de eficiência em modelos de regressão é uma parte essencial do processo de desenvolvimento de modelos de aprendizado de máquina. Essas métricas fornecem uma visão abrangente da qualidade do modelo e permitem tomar decisões informadas com base em previsões precisas. No Projeto utilizamos:

O **RMSE** (Root Mean Square Error) é uma métrica que quantifica a precisão de um modelo de previsão em relação aos valores reais. Ele calcula a raiz quadrada da média dos quadrados das diferenças entre as previsões e os valores reais. O RMSE fornece uma medida do desvio padrão dos erros do modelo, em uma escala de interpretação similar à unidade original dos dados. Um valor menor de RMSE indica um melhor ajuste do modelo aos dados e uma maior precisão nas previsões.

**METODOLOGIA**

Foi utilizada a base de dados “*Book-Crossing: User review ratings”*, disponível no *Kaggle*, na qual são apresentados conjuntos de dados que relacionam livros e avaliações de leitores. No contexto deste trabalho, entende-se por dados as principais características dos livros (ID\_LIVRO, TITULO, AUTOR) e dos leitores (ID\_USUARIO, ID\_LIVRO, AVALIACAO).

Para realizar a análise de dados, criar modelos de aprendizado de máquina e avaliar seu desempenho, foram utilizadas as seguintes bibliotecas Python:

* Matplotlib: para criação gráficos e visualizações de dados, ajudando na interpretação e comunicação de resultados;
* Numpy: para operações numéricas, incluindo manipulação de arrays multidimensionais;
* Pandas: para realização de operações como leitura, filtragem, agregação e transformação de dados;
* Profile Report: ferramenta do *pandas\_profiling* que gera um relatório detalhado sobre um *DataFrame* do pandas, fornecendo insights estatísticos e visuais sobre seus dados.
* Scikit-learn (também conhecida como *sklearn*): para implementações eficientes de uma ampla gama de algoritmos de aprendizado de máquina, incluindo *Nearest Neighbors*;
* Scipy: para criação de uma matriz esparsa (*sparse matrix*) a partir de um DataFrame do pandas; e
* Seaborn: para visualização de dados estatísticos e gráficos informativos.

Através da técnica K-Nearest Neighbors (KNN), o algoritmo KNN um método não paramétrico que se baseia na proximidade dos exemplos de treinamento para tomar decisões de classificação ou regressão. Basicamente esse algoritmo identifica através da similaridade para qual classe um ponto/dado pertence. Dentro do ambiente Machine Learning, o algoritmo de KNN é considerado um algoritmo de fácil entendimento, muitas vezes aplicado para reconhecimento de padrões, detectação de fraude e, como o caso apresentado, para sistemas de recomendações.

**ANÁLISE EXPLORATÓRIA**

A base de dados **BX\_Books** (“Livros”) contém informações sobre as obras (livros), composto por código do livro, título da obra, autor da obra, Ano da publicação da obra, editora e outras três variáveis com link da imagem do livro, cada coluna indica um tamanho da imagem, podemos entendo S-Small (pequeno), M-Medium (médio) e L-Large (grande).

* ISBN = International Standard Book Number (Número Padrão Internacional de Livro): Variável contendo um padrão numérico criado com o objetivo de fornecer um número de identificação para cada publicação monográfica;
* Book-Title = Título do livro: Variável que informa o nome da obra/livro;
* Book-Author = Nome do autor: Coluna informando o nome do autor responsável pela obra;
* Year-Of-Publication = Ano da publicação: Dados referente ao ano em que o livro/obra foi publicado;
* Publisher = Editora: Nome da editora responsável pela obra/livro;
* Image-URL-S = URL da imagem do livro tamanho pequeno;
* Image-URL-M = URL da imagem do livro tamanho médio; e
* Image-URL-L = URL da imagem do livro tamanho grande.

Texto

Descrição gerada automaticamente

A base de dados **BX-Books-Ratings** (“avaliações”) contém as avaliações dos usuários sobre as obras (livros), composto por Identificação do usuário, código do livro e avaliação do livro.

* User-ID = ID-Usuario: Variável contendo um código identificador de cada usuário;
* ISBN = International Standard Book Number (Número Padrão Internacional de Livro): Variável contendo um padrão numérico criado com o objetivo de fornecer um número de identificação para cada publicação monográfica; e
* Book-Rating = Classificação do Livro: Variável responsável pelos dados de avaliação do usuário (User-ID) sobre as obras (ISBN).

Texto

Descrição gerada automaticamente

Já a base de dados BX\_Users (“usuários”) contém informações sobre os usuários Amazon (clientes), contendo o User-ID, Localidade do usuário e idade.

* User-ID = ID-Usuário: Variável contendo um código identificador de cada usuário;
* Location = Localização: Estado onde o usuário está localizado; e
* Age = Idade: Idade do usuário.

Texto

Descrição gerada automaticamente

Para a melhor avaliação dos dados da base de dados, foram feitos os gráficos:

Contagem das cidades com maior quantidade de usuários:

Ícone

Descrição gerada automaticamente com confiança média

Distribuição dos usuários por idade:

Gráfico, Histograma

Descrição gerada automaticamente

Avaliação dos livros:

Interface gráfica do usuário

Descrição gerada automaticamente

**PRÉ-PROCESSAMENTO E LIMPEZA DOS DADOS**

Inicialmente, foram realizadas operações para filtrar e renomear colunas no DataFrame livros. Primeiro, foram selecionadas apenas as colunas necessárias, que são 'ISBN', 'Book-Title' e 'Book-Author'. Em seguida, as colunas foram renomeadas para 'ID\_LIVRO', 'TITULO' e 'AUTOR', respectivamente, usando o método 'rename' com um dicionário de mapeamento de nomes de colunas. O parâmetro 'inplace' foi definido como “True” para aplicar as mudanças diretamente ao DataFrame livros.

Interface gráfica do usuário, Texto, Aplicativo, chat ou mensagem de texto

Descrição gerada automaticamente

De forma semelhante, no DataFrame avaliações, foram selecionadas apenas as colunas necessárias, que são ‘User\_ID’, ‘ISBN’ e ‘Book-Rating’ que foram renomeadas ID\_USUARIO', 'ID\_LIVRO' e 'AVALIACAO’, respectivamente. Da mesma forma, o DataFrame avaliações foi ajustado.

Tabela

Descrição gerada automaticamente

Considerando a grande quantidade de avaliações classificadas como com nota zero (716.109 observações), essas instâncias foram excluídas e o DataFrame avaliações foi novamente ajustado, resultando em uma nova base de dados com 433.671 avaliações não nulas.

Tabela

Descrição gerada automaticamente

Foi realizada a contagem de avaliações por ID de livro, com a criação de uma Coluna de quantidade de avaliações no DataFrame ‘avaliacoes’ para contar o número de avaliações em cada grupo. O resultado foi armazenado em um novo DataFrame chamado 'contagem\_avaliacoes'. A função 'reset\_index()' foi usada para redefinir o índice do DataFrame, e o nome da coluna de contagem foi definido como 'QTDE\_AVALIACOES'.

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

Foram verificadas a existência de valores nulos dos DataFrames ‘livros’ e ‘avaliacoes’ e essas observações foram excluídas da base de dados, resultando em uma base de dados de 149.841 livros e 77.805 avaliações.

Posteriormente, nos DataFrames ‘livros’ e ‘avaliacoes’ foram selecionados apenas as avaliações cujos clientes tenham realizado mais de 9 avaliações, resultando em 5.444 observações livros e 295.561 observações em avaliações:

Texto

Descrição gerada automaticamente

Para a concatenação dos DataFrames, precisamos remover as letras do ‘ID\_LIVRO’ de ambas as bases de dados.

Texto

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente

Foi realizada a concatenação dos DataFrames considerando o ‘ID\_LIVROS’ para tal concatenação:

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

Ainda para a limpeza dos dados, foi realizada a verificação e exclusão de dados duplicados para que houvesse problemas de termos o mesmo usuário avaliando o mesmo livro mais de uma vez. Nossa base de dados resultante ficou com 88.579 observações.

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

**SISTEMA DE RECOMENDAÇÃO DE LIVROS**

Foi realizado um processo de pivoteamento (ou pivot) dos dados. A função ‘pivot\_table()’ foi utilizada para reorganizar o DataFrame ‘avaliacoes\_e\_livros’ de modo que cada ‘ID\_USUARIO’ passasse a ser uma coluna, e o valor de nota ‘AVALIACAO’ para cada livro avaliado fosse disposto em células correspondentes. A coluna ‘TITULO’ foi utilizada como índice para as linhas do DataFrame resultante.

Texto, Aplicativo

Descrição gerada automaticamente

Os valores nulos foram substituídos por zero para que todas as células tenham um valor numérico válido, facilitando operações e cálculos subsequentes.

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

O DataFrame livros\_pivot é passado como argumento para a função csr\_matrix() do Scipy, que o converte em uma matriz esparsa no formato CSR.

Uma matriz esparsa é uma matriz que possui a maioria de seus elementos iguais a zero e permite uma representação eficiente de conjuntos de dados, economizando dessa forma memória e operações.

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

**Modelo 1**

Foi implementado o modelo de vizinhos mais próximos (KNN) com a utilização do método "brute” para calcular as distâncias entre os pontos. Em seguida, o modelo foi treinado com os dados contidos na matriz esparsa.

Interface gráfica do usuário, Texto, Aplicativo, chat ou mensagem de texto

Descrição gerada automaticamente

Para este tipo de modelo preditivo de sistema de recomendação não foram separados dados de treino e de teste, simplesmente foi utilizado o KNN para fazer as recomendações por aproximação da distância. Nesse caso, uma forma de mensuração da "acurácia" seria verificar a menor distancia encontrada entre os pontos.

O que o KNN faz é trazer os livros avaliados positivamente pelos mesmos usuários. Ex, o livro 'X', 'Y' e 'Z' foram bem avaliado por Fulano e Beltrano, e Ciclano avaliou bem o filme 'Y', assim como o Fulano e o Beltrano fizeram. Significa que o algoritmo vai indicar como sugestões ao Ciclano os livros 'X' e 'Z'.

Foi realizado um processo de recomendação de livros com base no modelo treinado ‘modelo’. O livro de interesse no exemplo foi o "The Boy Next Door". Primeiro, foi feita uma filtragem do DataFrame ‘livros\_pivot’ usando o método ‘filter()’, selecionando apenas as informações referentes ao livro "The Boy Next Door".

Essas informações foram passadas para o método ‘kneighbors()’ do modelo treinado, que retornou as distâncias e as sugestões dos livros mais similares ao livro fornecido. O resultado é armazenado nas variáveis ‘distances’ e ‘sugestions’. Em seguida, um loop foi utilizado para iterar sobre as sugestões e imprimir os títulos dos livros similares, acessando os índices do DataFrame ‘livros\_pivot’ através dos índices armazenados em ‘sugestions’.

**Modelo 2**

Com o objetivo de construirmos um método de avaliação, a divisão entre conjunto de treinamento e teste foi adotada para permitir uma avaliação mais precisa do desempenho do sistema de recomendação KNN. Esta abordagem permite uma estimativa realista de sua eficácia em um ambiente de produção.

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

Posteriormente, foi realizado o cálculo das métricas para avaliar o desempenho do modelo de previsão. Medidas essas, essenciais para avaliar a qualidade do modelo.

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

* Erro quadrático médio (RMSE) – Cálculo entre as previsões e os valores reais do conjunto de teste. Este valor é obtido através da função ‘mean\_squared\_error’ do pacote ‘numpy’. O código então converte as previsões e os valores reais em valores binários, utilizando um limite de 3.5 como ponto de corte. Isso significa que qualquer valor de previsão acima de 3.5 será considerado como 1, indicando uma classificação positiva, enquanto valores iguais ou abaixo de 3.5 serão considerados como 0.
* Precisão, recall e F1-score, utilizando as funções ‘precision\_score’, ‘recall\_score’ e ‘f1\_score’, do pacote ‘sklearn.metrics’. Essas métricas ajudam a avaliar a capacidade do modelo de prever corretamente as classificações binárias.

**AVALIAÇÃO DE DESEMPENHO**

Desenvolvemos dois modelos de recomendação KNN para fins de comparação e avaliação.

**Modelo 1**

Utilizou-se da base completa para treinamento e realizou as recomendações dos livros com base na similaridade verificada por meio das distâncias.

Como exemplo, foram selecionados alguns livros para verificar a recomendação sugerida:

* **The Boy Next Door**

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

* **Artemis Fowl (Artemis Fowl, Book 1)**

Interface gráfica do usuário

Descrição gerada automaticamente com confiança baixa

* **Hoot (Newbery Honor Book)**

Uma imagem contendo Texto

Descrição gerada automaticamente

Como no modelo não houve a separação entre a base de treinamento e teste, uma possibilidade de mensuração da “acurácia” seria através do cálculo das distâncias entre os pontos.

**Modelo 2**

Foi especificamente projetado para calcular métricas de classificação, como precisão, recall e F1-score. Após a execução do modelo preditivo, foram aplicadas as métricas de desempenho usando os dados de teste, fornecendo uma estimativa geral do desempenho.

Dados obtidos:

RMSE: 0.4140654415775425

Precisão: 0.093904924014252

Recall: 0.02953036788882685

F1-score: 0.03532176934974752

Os resultados revelam que o desempenho do modelo de recomendação está aquém do ideal, com um RMSE de 0.41 indicando um desvio significativo nas previsões. Este valor indica que as recomendações feitas pelo modelo podem estar distantes das preferências reais dos usuários, sugerindo a necessidade de melhorias na precisão das previsões.

Além disso, a baixa precisão de 9% e recall de 3% revelam que apenas uma pequena proporção das recomendações feitas pelo modelo é relevante para os usuários, resultando em uma pontuação F1-score de 0.035. Esta pontuação reflete um desequilíbrio entre precisão e recall, sugerindo que o modelo pode estar falhando em capturar adequadamente as preferências dos usuários e em fornecer recomendações relevantes.

**CONCLUSÃO E TRABALHOS FUTUROS**

Neste trabalho, foi desenvolvido um projeto de recomendação de livros utilizando a base de dados "Book-Crossing: User review ratings" do Kaggle e técnicas de aprendizado de máquina, focando no algoritmo K-Nearest Neighbors (KNN). Inicialmente, foi discutida a importância dos sistemas de recomendação na era da informação, destacando sua relevância na filtragem e descoberta de conteúdo relevante. Em seguida, foram apresentados o referencial teórico e a metodologia adotada, que incluiu a exploração da base de dados, a implementação dos modelos de recomendação KNN e a avaliação de seu desempenho.

O objetivo principal deste trabalho era aprimorar a precisão das recomendações de livros e adaptá-las às preferências individuais dos usuários. O primeiro modelo nos forneceu insights valiosos sobre as previsões e o segundo modelo nos permitiu analisar a capacidade do sistema de recomendação em identificar corretamente as preferências dos usuários.

Embora tenham sido obtido resultados promissores com a implementação dos modelos de KNN, observou-se que o desempenho ainda está aquém do ideal, enquanto para o primeiro obteve-se dificuldade de mensuração da “acurácia”, para o segundo foi identificado desvio significativo nas previsões e baixa precisão e recall.

Esses resultados indicam a necessidade de melhorias nas previsões e na capacidade de capturar as preferências dos usuários. Uma possibilidade seria a exploração de técnicas mais avançadas de aprendizado de máquina, como redes neurais profundas (*deep learning*), que podem oferecer maior poder de generalização e captura de padrões complexos nos dados. Além disso, seria interessante considerar a inclusão de outras fontes de dados, como informações demográficas dos usuários ou dados de interações sociais, para enriquecer as recomendações e torná-las ainda mais personalizadas.

Além disso, as técnicas e metodologias utilizadas neste projeto podem ser aplicadas em diferentes contextos e tipos de recomendação, não se limitando apenas a livros. Futuros trabalhos podem explorar a aplicação dessas técnicas em áreas como recomendação de filmes, músicas, produtos de comércio eletrônico, entre outros. Seria interessante também investigar abordagens híbridas que combinam diferentes algoritmos de recomendação para obter resultados ainda mais precisos e diversificados. Essas direções futuras representam oportunidades para avançar no campo dos sistemas de recomendação e proporcionar experiências mais personalizadas e relevantes para os usuários.

**BIBLIOGRAFIA**

CHAPMAN, P.et al.CRISP-DM 1.0: Step-by-step data mining guide.SPSS inc, v. 9, p. 13, 2000. Disponível em: <https://www.kde.cs.uni-kassel.de/wp-content/uploads/lehre/ws2012-13/kdd/files/CRISPWP-0800.pdf>. Acesso em: 31 mar. 2024.

GOLDSCHMIDT, Ronaldo. Data Mining. [Digite o Local da Editora]: Grupo GEN, 2015. E-book. ISBN 9788595156395. Disponível em: <https://app.minhabiblioteca.com.br/#/books/9788595156395/>. Acesso em: 30 mar. 2024.

GUIDO, A. C. S. Introduction to Machine Learning with Python. [S.l.]: O’Reilly Media, 2016.

IBM SPSS. IBM SPSS modeler text analytics 16 user guide. 2016. Disponível em: <https://public.dhe.ibm.com/software/analytics/spss/documentation/modeler/16.0/en/ta_guide_book.pdf>. Acesso em: 29 mar. 2024.

MEDEIROS, I. Estudo sobre sistemas de recomendação colaborativos. [S.l.]: Recife, 2013.

REZENDE, S. O. Sistemas inteligentes: fundamentos e aplicações. [S.l.]: Manole, 2005.

SHEARER, C. The CRISP-DM Model: The New Blueprint for Data Mining. Journal of Data Warehousing, v. 5, n. 4, 2000. Disponível em: [https://www.academia.edu/42079490/CRISP\_DM\_The\_New\_Blueprint\_for\_Data\_Mining\_Colin\_Shearer\_Fall\_2000](https://www.academia.edu/42079490/CRISP_DM_The_New_Blueprint_for_Data_Mining_Colin_Shearer_Fall_2000_). Acesso em: 29 mar. 2024.