

Universidade Presbiteriana Mackenzie  
Faculdade de Computação e Informática – FCI

## PROJETO APLICADO III

### Sistemas de Recomendação

#### COMPONENTES DO GRUPO:

Carlos Antonio Batista	- TIA 22021477
Erick Meyer Machado Terceiro	- TIA 22008225
Mauricio Henrique Leal Novakowski	- TIA 22015078
Pedro Costa Dias	- TIA 22010823

São Paulo

2023

## SUMÁRIO:

### 1 - Introdução:

1.1 - Contexto do trabalho .....	3
1.2 – Motivação-Justificativas .....	5
1.3 - Objetivos .....	6
1.4 – Cronograma e Diretório .....	6

### 2 - Referencial Teórico .....

2.1 – Descrição do Projeto .....	11
2.1.1 – Definição da linguagem usada .....	11
2.1.2 – Análise exploratória .....	11
2.1.3 – Definição da técnica de treinamento do modelo .....	12
2.1.4 – Treinamento do modelo .....	13
2.1.5 – Definição da forma de avaliação do desempenho do modelo .....	13

### 3 - Metodologia .....

### 4 - Resultados .....

### 5 - Conclusão e Trabalhos Futuros .....

### 6 – Resumo .....

**Universidade Presbiteriana Mackenzie**  
**Faculdade de Computação e Informática – FCI**

## 1- INTRODUÇÃO

### 1.1 – Contexto do Trabalho:

Após várias pesquisas e análises de datasets no Kaggle e no repositório da Universidade da Califórnia, escolhemos o conjunto de dados “Receitas e interações do Food.com” para aplicarmos o Sistema de Recomendação, treinar e testar os métodos mais eficazes para gerar receitas personalizadas a partir do histórico de preferência do usuário (retirado do Kaggle).

Kaggle é uma plataforma para aprendizado de Ciências de Dados que possui inúmeros conjuntos de dados para todo o tipo de treinamento em Data Science e Machine Learning.

O repositório da Universidade da Califórnia possui uma infraestrutura técnica robusta, conectados a mais de 100 mil redes de pesquisa e educação de alto desempenho em todo o mundo. Esse repositório digital leva a segurança muito a sério e são implementadas medidas de proteção dos dados dos clientes.

Sistemas de recomendação são aplicações que conseguem sugerir algo a uma pessoa, com a ajuda de uma predição probabilística de que ele vai gostar daquilo. Envolve uma análise profunda que compreende padrões, correlações entre os dados e até mesmo a distância entre as variáveis existentes na base de dados.

Link dataset: <https://www.kaggle.com/datasets/shuyangli94/food-com-recipes-and-user-interactions>

**Universidade Presbiteriana Mackenzie**  
**Faculdade de Computação e Informática – FCI**

Fonte da base de dados:

<b>Forma de coleta</b>	Raspado por meio de solicitações Python/BeautifulSoup
<b>Início da coleta</b>	24/02/2000
<b>Término da coleta</b>	17/12/2018
<b>Origem dos dados</b>	Food.com
<b>Proprietário da base</b>	Shuyang Li
<b>Editor primário da base</b>	Shuyang Li
<b>Editor secundário da base</b>	Bodhisattwa Prasad Majumder
<b>DOI</b>	10.34740/kaggle/dsv/783630
<b>Licença</b>	Arquivo de dados © Autores originais
<b>Frequência de atualização</b>	Atualizado há 4 anos

Fonte: Elaborado pelo próprio autor com dados do Kaggle.

**Universidade Presbiteriana Mackenzie**  
**Faculdade de Computação e Informática – FCI**

**1.2- Motivação - Justificativas:**

A relevância do tema é, a partir de receitas consumidas pelo usuário anteriormente, atender em nível de técnica e receita, aquela que seja plausível e personalizada para cada tipo de usuário, tornando-o mais feliz e satisfeito além de colaborar com a sua saúde alimentar e física.

Sabemos que uma alimentação saudável é fundamental para o bom funcionamento do organismo. A alimentação saudável aliada a exercícios físicos contribui para a qualidade de vida, melhorando o sistema imunológico, a capacidade de concentração, prevenindo doenças entre outros benefícios.

A má alimentação é uma das principais causas de mortes no mundo. A alimentação inadequada está relacionada ao desenvolvimento de doenças e problemas de saúde como: obesidade e sobrepeso, doenças cardiovasculares, diabetes entre outras.

Nesse contexto, esse projeto pode ajudar todas as pessoas a prepararem uma refeição simples e saudável de acordo com os gostos e preferências individuais, sem excessos e exageros, incentivando a boa prática alimentar, experimentando alimentos diferentes e tornando-os mais receptivos a determinados alimentos que não costumam comer.

**Universidade Presbiteriana Mackenzie**  
**Faculdade de Computação e Informática – FCI**

**1.3 – Objetivos:**

Este projeto tem por objetivo gerar receitas personalizadas para ajudar os usuários com preferências culinárias. Ajudar as pessoas a mudarem o seu comportamento alimentar, desenvolvendo sistemas capazes de recomendar receitas saudáveis e que levam em conta as necessidades e preferências (gostos) dos usuários e, também, a experimentar novos alimentos.

**1.4 – Cronograma e Diretório:**

<b>Período:</b>	<b>Atividade:</b>	<b>Agenda:</b>
01/08/23 a 27/08/23	<p style="text-align: center;"><b>Etapa 1</b></p> <ul style="list-style-type: none"> <li>• <input checked="" type="checkbox"/> <del>Organização dos grupos de trabalho.</del></li> <li>• <input checked="" type="checkbox"/> <del>Escolha do tema do projeto.</del></li> <li>• <input checked="" type="checkbox"/> <del>Organização do repositório de materiais.</del></li> <li>• <input checked="" type="checkbox"/> <del>Cronograma do projeto.</del></li> <li>• <input checked="" type="checkbox"/> <del>Seção: Capa</del></li> <li>• <input checked="" type="checkbox"/> <del>Seção: Sumário (Parcial)</del></li> <li>• <input checked="" type="checkbox"/> <del>Seção: Introdução</del></li> </ul>	21/08 Encontro com o Professor.
		21/08 Reunião 1 - Levantamento de dúvidas.
		23/08 Reunião 2 - Decisões finais e divisão de tarefas.
		25/08 Prazo para tarefas individuais.
		27/08 Prazo Final para Entrega

**Universidade Presbiteriana Mackenzie**

**Faculdade de Computação e Informática – FCI**

28/08/23 a 22/09/23	<p align="center"><b>Etapa 2</b></p> <ul style="list-style-type: none"> <li><input type="checkbox"/> Definir as bibliotecas Python.</li> <li><input type="checkbox"/> Analisar, de forma exploratória, a base de dados.</li> <li><input type="checkbox"/> Tratar e preparar a base de dados para o treinamento.</li> <li><input type="checkbox"/> Definir a técnica para o treinamento do modelo de recomendação.</li> <li><input type="checkbox"/> Realizar o treinamento de um modelo inicial, como prova de conceito.</li> <li><input type="checkbox"/> Definir a forma de avaliação de desempenho do modelo.</li> <li><input type="checkbox"/> Descrever o referencial teórico para a elaboração do projeto.</li> <li><input type="checkbox"/> Seção: Referencial Teórico</li> <li><input type="checkbox"/> Seção: Sumário (Update)</li> </ul>	28/08 Reunião 1 - Levantamento de dúvidas.
		05/08 Reunião 2 - Decisões finais e divisão de tarefas.
		08/08 Análise Exploratória.
		11/08 Treinamento do modelo.
		17/08 Prazo para tarefas individuais.
		18/08 Encontro com o Professor.
		22/09 Prazo Final para Entrega.
23/09/23 a 23/10/23	<p align="center"><b>Etapa 3</b></p> <ul style="list-style-type: none"> <li><input type="checkbox"/> Analisar os resultados preliminares alcançados na etapa anterior.</li> <li><input type="checkbox"/> Ajustar o pipeline de treinamento para melhoria do desempenho do modelo.</li> <li><input type="checkbox"/> Reavaliar o desempenho do modelo.</li> <li><input type="checkbox"/> Organizar, de forma sistemática, a descrição das técnicas utilizadas.</li> <li><input type="checkbox"/> Descrever a metodologia aplicada no projeto.</li> <li><input type="checkbox"/> Seção: Metodologia</li> <li><input type="checkbox"/> Seção: Sumário (Update)</li> </ul>	25/09 Reunião 1 - Levantamento de dúvidas.
		27/09 Reunião 2 - Decisões finais e divisão de tarefas.
		??/09 Encontro com o Professor.
		20/10 Prazo para tarefas individuais.
		23/10 Prazo Final para Entrega

**Universidade Presbiteriana Mackenzie**  
**Faculdade de Computação e Informática – FCI**

24/10/23 a 1/11/23	<p style="text-align: center;"><b>Etapas 4</b></p> <ul style="list-style-type: none"> <li><input type="checkbox"/> Organizar os resultados alcançados.</li> <li><input type="checkbox"/> Analisar os resultados, identificando pontos positivos e negativos das técnicas utilizadas.</li> <li><input type="checkbox"/> Descrever e documentar os resultados.</li> <li><input type="checkbox"/> Descrever e documentar as conclusões e os trabalhos futuros.</li> <li><input type="checkbox"/> Entregar a apresentação em vídeo do projeto.</li> <li><input type="checkbox"/> Entregar os artefatos de software.</li> <li><input type="checkbox"/> Seção: Resultados</li> <li><input type="checkbox"/> Seção: Conclusão e trabalhos futuros</li> <li><input type="checkbox"/> Seção: Resumo</li> <li><input type="checkbox"/> Seção: Sumário (Update final)</li> <li><input type="checkbox"/> Entregar a documentação do projeto.</li> </ul>	25/10 Reunião - Levantamento de dúvidas, decisões finais e divisão das tarefas.
		??/10 Encontro com o Professor.
		28/10 Prazo para tarefas individuais.
		01/11 Prazo Final para Entrega

Fonte: Elaborado pelo próprio autor.

Link para o diretório: <https://github.com/PedroCosDi/ProjetoAplicadoMack3>



**Universidade Presbiteriana Mackenzie**  
**Faculdade de Computação e Informática – FCI**

## 2 – REFERENCIAL TEÓRICO:

Projeto de conclusão de curso da Universidade Federal da Fronteira Sul – Campos Chapecó – Ciência da Computação de 2022 sobre sistema de recomendação de receitas alimentares utilizando filtragem baseada em conteúdo.

O projeto trata da dificuldade, nos últimos anos, de se manter uma alimentação saudável devido a fatores como reflexo da pandemia, problemas psicológicos como ansiedade e depressão.

A internet nos traz uma quantidade de receitas alimentares disponíveis e que vem crescendo a cada dia. Por conta disso, o projeto propõe um desenvolvimento de um sistema de recomendação de receitas, onde o usuário transforma o ato de cozinhar em um hobby e a necessidade de manter uma vida mais saudável. O projeto usa a filtragem baseada em conteúdo para fazer a recomendação e que também é feita por meio de cinco receitas por maior similaridade utilizando a distância euclidiana.

A filtragem baseada em conteúdo seleciona itens considerando o conteúdo de cada um e comparando ao conteúdo dos itens previamente avaliados pelo usuário, e que parece ser a mais apropriada para o projeto.

O objetivo do desenvolvimento desse projeto é possibilitar a melhor compreensão dos métodos de extração de dados, ou web scraping, além de tratar e padronizar os dados que poderão ser utilizados como facilitador no desenvolvimento de projetos futuros.

No que se refere aos algoritmos de aprendizado de máquina, são responsáveis por realizar tarefas desejadas de modo que produzam uma saída esperada; algoritmos esses que funcionam por meio de repetição, modo em que as saídas ficam mais precisas a cada repetição feita. Para que isso ocorra, o conjunto de dados foi dividido em treino e teste.

No projeto foi usado o KMeans que é um algoritmo de clusterização, processo de classificação de padrões na forma não supervisionada em grupos ou clusters. Para isso foi necessário determinar o número de clusters e o centro de cada um deles.

**Universidade Presbiteriana Mackenzie**  
**Faculdade de Computação e Informática – FCI**

Utilizando as bibliotecas existentes para a linguagem de programação Python, a experimentação demonstrou que, no conjunto de dados extraído de um site chamado “Tudo Gostoso”, obteve-se um melhor resultado utilizando o algoritmo KMeans com três clusters.

Também possibilitou a verificação das recomendações usando a distância euclidiana entre os objetos como medida de similaridade.

A ferramenta teve um resultado razoável em relação às recomendações geradas.

Para trabalhos futuros, pensam em gerar um sistema de recomendação mais preciso, com receitas que possuam ingredientes mais específicos e também desenvolver um sistema capaz de aperfeiçoar as recomendações, criando perfis, separados por grupos, condizentes ao gosto do usuário.

Link do trabalho de conclusão de curso

<https://rd.uffs.edu.br/bitstream/prefix/5761/1/BEVILACQUA.pdf>

**Universidade Presbiteriana Mackenzie**  
**Faculdade de Computação e Informática – FCI**

**2.1 - Descrição do Projeto:**

**2.1.1 – Definição da Linguagem de Programação Usada:**

A linguagem de programação usada no projeto será o Python, linguagem de alto nível, sintaxe relativamente simples e de fácil compreensão. Possui um grande número de bibliotecas, dentre elas usaremos pandas, numpy, seaborn, matplotlib, sklearn, csr\_matrix(scipy,sparse), entre outros.

**2.1.2 – Análise Exploratória:**

Esse projeto tem como objetivo sistema de recomendação de usuários de “Receitas e Interações do Food.com” para gerar receitas personalizadas, ajudando o usuário com suas preferências culinárias, mudando seu comportamento alimentar levando em consideração as necessidades e preferências e, o mais importante, experimentar novos alimentos.

Para a análise exploratória, que consiste em toda e qualquer operação realizada com os dados com o objetivo de garantir uma base de dados limpa e organizada para facilitar a compreensão dos dados. Foi realizado o tratamento da base de dados com a limpeza dos dados, verificando e substituindo valores ausentes (dropando esses valores – pois quando encontrado um valor em branco, o nosso modelo preditivo poderá dar erro) ,contagem de linhas e colunas, exclusão de colunas desnecessárias para o projeto, verificação e tratamento de valores duplicados tanto para o dataset receitas ,como para o dataset interações que serão agrupados. Foi verificado os tipos de variáveis de ambos datasets. Depois do agrupamento, será novamente verificado o número de linhas e colunas, removemos a duplicidade de usuários que fizeram review de receitas duas ou mais vezes, verificamos novamente os valores duplicados para que não tenhamos problemas de um usuário avaliar a mesma receita diversas vezes.

Foi removido o “ ID\_RECEITA” para que a receita seja recomendada pelo “NOME\_RECEITA”.

**Universidade Presbiteriana Mackenzie**  
**Faculdade de Computação e Informática – FCI**

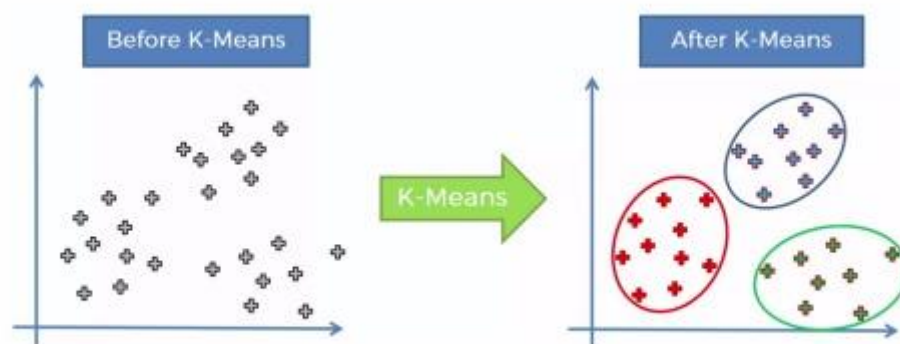
Criamos um novo dataset com as receitas em linhas e os usuários em colunas, fazendo um pivô para que cada “ID\_USUARIO” seja uma variável com o respectivo valor de avaliação para cada receita avaliada. Importamos o `csr_matrix` (pacote `Scipy`) que é um método para criar a matriz esparsa, que é uma matriz na qual a grande maioria de seus elementos possui um valor padrão, que no caso do nosso projeto, esse valor é o zero. Assim, foi criado e testado o modelo K Vizinhos Mais Próximos (KNN) e as recomendações foram realizadas.

### 2.1.3 – Definição da técnica de treinamento do modelo de recomendação:

A técnica de treinamento do modelo de recomendação utilizada nesse projeto será o K Vizinhos Mais Próximos (KNN), modelo bem simples de ser compreendido, apenas é necessário que tenha uma noção de distância e que quanto maior é a proximidade dos pontos, maior é a similaridade entre eles.

Para fazer a recomendação, o filtro colaborativo e o filtro por conteúdo busca por usuários com dados similares e a partir desses dados faz a recomendação de itens ainda não consumidos pelo usuário que está recebendo a recomendação cruzada com os gostos de outro usuário com perfil parecido, com isso se evita que o usuário fique dentro de uma bolha de conteúdos, mas que tem que ser medido para não diversificar demais o conteúdo e o usuário não se identificar com o que foi apresentado.

O KNN faz a predição da vizinhança K do usuário solicitante da recomendação com base em suas classificações positivas ou negativas, para então fazer a previsão das avaliações do usuário.



Fonte: médium (2020)

**Universidade Presbiteriana Mackenzie**  
**Faculdade de Computação e Informática – FCI**

#### 2.1.4 – Treinamento do Modelo:

O sistema de filtro colaborativo usa as ações dos usuários para recomendar outras receitas. Já o sistema de filtro por conteúdo usa uma lista de receitas inspirada pelo histórico de navegação do usuário (baseada nas últimas receitas adquiridas pelo usuário).

Logo, o filtro precisa conhecer o domínio em que ele atua de forma que consiga fazer as comparações e retornar receitas semelhantes.

O algoritmo ideal para se utilizar em um sistema de recomendação, são os algoritmos que se calculam e que se baseiam em distância, algoritmo esse que é o K Vizinhos Mais Próximos (KNN) que é um excelente algoritmo quando se deseja fazer modelos preditivos referente a cálculos de distância.

Como cada usuário virou uma variável, para criar o modelo preditivo usaremos o conceito de matriz esparsa, uma matriz com muitos valores zero e que conseguimos compactar essa matriz, e onde houver valores zero, haverá uma função que irá gravar essas posições de zeros.

Para fazer uma conversão para uma matriz esparsa é bem simples, ou seja, no pacote scipy utilizar método `csr_matrix` e a partir daí já podemos criar o nosso modelo preditivo com uma matriz compactada.

#### 2.1.5 – Definição da Forma de Avaliação de Desempenho do Modelo:

A análise de grandes quantidades de dados torna-se uma tarefa desafiadora, e para essa tarefa é imprescindível a utilização de ferramentas computacionais que, de forma inteligente, processem as informações dos bancos de dados para auxiliarem na tomada de decisão.

O objetivo desse projeto é buscar o melhor resultado do desempenho do algoritmo K Vizinhos Mais Próximos (KNN) no sistema de recomendação.

Para avaliar o desempenho dos algoritmos é fundamental entender as metodologias de avaliação através da comparação das predições realizadas e as respectivas avaliações reais de usuários para as instâncias preditas.

A obtenção de métricas para aferir o desempenho de um sistema de recomendação é fundamental para verificarmos se as predições realizadas são adequadas para o projeto.

**Universidade Presbiteriana Mackenzie**  
**Faculdade de Computação e Informática – FCI**

Após importação do algoritmo KNN do Scikit Learn, iremos criar e treinar nosso modelo preditivo. Para isso criamos um objeto chamado “modelo” que receberá o pacote do KNN (NearestNeighbors) usando também um hiperparâmetro chamado “algorithm = brute”, que é um método direto de resolver um problema que depende de poder computacional puro e de tentar todas as possibilidades em vez de técnicas avançadas para melhorar a eficiência.

O “modelo.fit” é o comando para se fazer o treino do modelo preditivo. Ao executarmos, ele criará e treinará nosso modelo.

A partir daí é só realizar as previsões sugerindo ao usuário as receitas. O Kneighbors é o que irá retornar de previsão dos vizinhos mais próximos (que são as recomendações) através da distância e das sugestões.

E por fim, os parâmetros é quem irão ser passados para o nosso modelo fazer as previsões.

