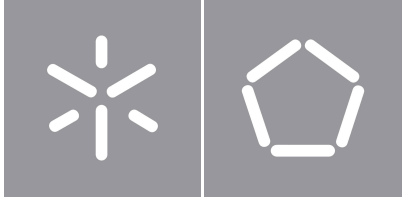
Universidade do Minho
Escola de Engenharia

Gonçalo Costa de Almeida

O poder da Inteligência Artificial no e-commerce

abril 2024



Universidade do Minho
Escola de Engenharia

Gonçalo Costa de Almeida

O poder da inteligência artificial no e-commerce

Dissertação de Mestrado
Mestrado em Engenharia Informática

Trabalho efetuado sob a orientação de
Cesar Analide

abril 2024

Direitos de Autor e Condições de Utilização do Trabalho por Terceiros

Este é um trabalho académico que pode ser utilizado por terceiros desde que respeitadas as regras e boas práticas internacionalmente aceites, no que concerne aos direitos de autor e direitos conexos.

Assim, o presente trabalho pode ser utilizado nos termos previstos na licença abaixo indicada.

Caso o utilizador necessite de permissão para poder fazer um uso do trabalho em condições não previstas no licenciamento indicado, deverá contactar o autor, através do RepositóriUM da Universidade do Minho.

Licença concedida aos utilizadores deste trabalho:

[Caso o autor pretenda usar uma das licenças Creative Commons, deve escolher e deixar apenas um dos seguintes ícones e respetivo lettering e URL, eliminando o texto em itálico que se lhe segue. Contudo, é possível optar por outro tipo de licença, devendo, nesse caso, ser incluída a informação necessária adaptando devidamente esta minuta]

CC BY

https://creativecommons.org/licenses/by/4.0/ *[Esta licença permite que outros distribuam, remixem, adaptem e criem a partir do seu trabalho, mesmo para fins comerciais, desde que lhe atribuam o devido crédito pela criação original. É a licença mais flexível de todas as licenças disponíveis. É recomendada para maximizar a disseminação e uso dos materiais licenciados.]*

Agradecimentos

Escrevo este excerto para expressar a mais profunda gratidão por quem me acompanhou nesta longa jornada, que termina com a apresentação da minha tese de mestrado. Este momento encerra um capítulo de cinco anos na minha vida, um período que será recordado com carinho. Durante este período desafiante, fui abençoado com o apoio inestimável de pessoas notáveis e aproveito esta oportunidade para expressar o meu sincero agradecimento.

Primeiramente, quero agradecer à Universidade do Minho por fornecer as ferramentas essenciais para completar este curso exigente e aos professores que moldaram o meu caminho académico. Em particular, ao Professor Analide, que não só orientou, mas também motivou-me a entregar o melhor de mim durante este percurso. Quero também estender o meu agradecimento aos meus colegas de turma e aos amigos de infância. Estiveram ao meu lado nos momentos mais difíceis, fornecendo apoio inabalável e ajudando-me a ultrapassar obstáculos que, sem eles, teriam parecido intransponíveis.

À empresa Bsolus, que generosamente me acolheu neste projeto, expresso a minha sincera gratidão. Nada teria sido possível sem a camaradagem, amizade e excelentes condições de trabalho que encontrei no local de trabalho. Quero dedicar um agradecimento especial à minha equipa na Bsolus: Dener, Miguel, Amaro, Fábio e José. Vocês não só me ensinaram muito, mas também sempre houve um enorme esforço para me integrar da melhor forma possível. Esta experiência foi enriquecedora em todos os aspetos.

Por último, mas não menos importante, quero expressar a minha eterna gratidão aos meus pais, Elisabete Costa e Clemente Almeida. Foram vocês que moldaram o homem que sou hoje. O vosso sacrifício, amor, compaixão e respeito não só moldaram o meu carácter, mas também a minha jornada académica. Sem o vosso apoio incondicional, nada disto serial possível.

A todos os que, de alguma forma, contribuíram para o meu sucesso, o mais sincero agradecimento. Este momento de realização é compartilhado com cada um de vocês, que este seja um marco numa jornada cheia de vitórias.

Declaração de Integridade

Declaro ter atuado com integridade na elaboração do presente trabalho académico e confirmo que não recorri à prática de plágio nem a qualquer forma de utilização indevida ou falsificação de informações ou resultados em nenhuma das etapas conducente à sua elaboração.

Mais declaro que conheço e que respeitei o Código de Conduta Ética da Universidade do Minho.

Universidade do Minho, Braga, abril 2024

Gonçalo Costa de Almeida

Resumo

O comércio eletrónico, igualmente conhecido por *e-commerce*, é o processo de compra e venda de produtos e serviços através da *world wide web*. O *e-commerce* tem um peso considerável no nosso quotidiano, sendo regularmente utilizado por diversas organizações para inúmeras transações. Perante este facto, tudo indica que o futuro do setor *e-commerce* é promissor e propicio a uma crescente integração de inteligência artificial nos serviços disponíveis à sociedade. Esta integração promete melhorias para as duas extremidades da conversa, tanto para o vendedor como para o cliente, desde inovações nos processos de compra a novas formas de atendimento ao cliente.

A essência desta tese reside na criação de um modelo de recomendações, construído com técnicas de inteligência artificial para lojas *e-commerce*. Concretamente, a utilização de inteligência artificial para a recomendação de produtos aos clientes aderentes à plataforma. O objetivo é demonstrar como a aplicação do modelo de recomendações criado pode impulsionar as vendas ao incentivar clientes a comprar produtos especialmente recomendados, conforme as preferências e características do cliente. Num mundo ideal, o algoritmo desenvolvido prevê os produtos que o cliente mais necessita ou deseja e recomenda as mesmas.

Para tal, será desenvolvido um serviço constituído por sistemas de recomendação que utilizarão técnicas de inteligência artificial. O produto final permitirá a extração de dados de plataformas de comércio eletrónico, a sua manipulação e a utilização de algoritmos de *Machine Learning* capazes de gerar recomendações personalizadas com base nos gostos e preferências de cada cliente.

Com o intuito de demonstrar a viabilidade e eficácia deste projeto, serão utilizados dados reais da empresa DeltaQ Portugal, possibilitando uma análise prática às implicações que a aplicação de inteligência artificial em plataformas *e-commerce* possa ter. Este estudo visa fornecer informações valiosas para as plataformas aderentes, realçando o potencial da inteligência artificial como uma ferramenta estratégica para melhorar a satisfação do cliente e aumentar as suas vendas.

Palavras-chave *e-commerce*, *Machine Learning*, *Data Mining*,

Abstract

E-commerce is the process of buying and selling products and services via the *world wide web*. Without even realizing it, *e-commerce* has a considerable impact on our daily lives, and is widely used by various organizations daily for various transactions. Given this fact, everything indicates that the future of this sector is promising and, consequently, that artificial intelligence is increasingly being integrated into its services. This integration promises improvements at both ends of the conversation, for both the seller and the customer, from innovations in purchasing processes to new forms of customer service.

The essence of this thesis lies in creating a recommendation model, built using artificial intelligence techniques for *e-commerce* stores. Specifically, the use of artificial intelligence to recommend products to customers who are members of the platform. The aim is to demonstrate how the application of the recommendations model created can boost sales by encouraging customers to buy products that have been specially recommended to them according to their preferences and characteristics. In an ideal world, the algorithm developed would predict customers' future purchases and recommend them.

To complete this objective, a service will be developed consisting of recommendation systems using artificial intelligence techniques. The end product will allow for the extraction of data from e-commerce platforms, its manipulation and the use of *Machine Learning* algorithms capable of generating personalized recommendations based on the tastes and preferences of each customer.

In order to demonstrate the feasibility and effectiveness of this project, real data from the company DeltaQ Portugal will be used, enabling a practical analysis of the implications that the application of artificial intelligence in *e-commerce* platforms may have. This study aims to provide valuable information for participating platforms, highlighting the potential of artificial intelligence as a strategic tool for improving customer satisfaction and increasing sales.

Key-Words *e-commerce*, *Machine Learning*, *Data Mining*

Conteúdo

1 Introdução **1**

1.1 Contexto . . . 1

1.2 Motivação . . . 3

1.3 Objetivos . . . 4

1.4 Estrutura e Planeamento . . . 5

1.4.1 Estrutura . . . 5

1.4.2 Calendarização . . . 6

2 Estado da arte **8**

2.1 O comércio . . . 8

2.1.1 De comércio a *e-commerce* . . . 8

2.1.2 Relação entre e-commerce, plataformas e negócio-cliente . . . 9

2.2 Integração dos sistemas de recomendação . . . 10

2.2.1 Sistemas de Recomendação . . . 10

2.2.2 Integração de Inteligência Artificial . . . 11

2.2.3 O Caso de Sucesso . . . 11

2.3 Recomendações e Tratamento de Dados . . . 13

2.3.1 A Importância do Tratamento de Dados . . . 13

2.3.2 Filtragem Colaborativa . . . 14

2.3.3 Filtragem Baseada no Conteúdo . . . 15

2.3.4 Filtragem Híbrida . . . 16

2.4 Problemas e desafios . . . 17

2.4.1 Análise de dados . . . 17

2.4.2 *Cold-start* . . . 17

2.4.3 Escalabilidade . . . 18

2.4.4 Dispersão . . . 18
2.4.5 Super especialização . . . 19
2.4.6 Outros . . . 19

3 Abordagem Proposta **20**
3.1 Metodologia de Mineração de Dados . . . 20
3.2 Ferramentas de programação . . . 23
3.2.1 Linguagem de programação . . . 23
3.2.2 Ferramentas de programação . . . 24
3.3 Armazenamento de recomendações . . . 24
3.4 Arquitetura do Sistema . . . 26
3.4.1 Síntese . . . 26
3.4.2 Comunicação entre micro serviços . . . 27
3.4.3 Makefile . . . 28
3.4.4 Arquivo de variáveis . . . 28
3.5 Documentação . . . 30

4 Implementação **31**
4.1 Compreensão de dados . . . 31
4.1.1 Exploração de Dados: Análise Preliminar . . . 31
4.1.2 Contrato de dados: atributos necessários . . . 33
4.2 Preparação dos dados . . . 36
4.2.1 Redução de dados . . . 36
4.2.2 Pré-processamento de Dados . . . 37
4.2.3 Aumento de performance . . . 43
4.3 Modelação . . . 44
4.3.1 Popularidade . . . 45
4.3.2 Modelo Complementar . . . 45
4.3.3 Clustering . . . 46
4.3.4 Filtragem Baseada em Conteúdo . . . 48
4.3.5 Filtragem Colaborativa . . . 49
4.3.6 Modelo Híbrido . . . 50

5 Análise de Resultados **52**

5.1 Avaliação . . . 52
5.1.1 Divisão dos dados . . . 52
5.1.2 Métricas . . . 52
5.1.3 Variável evaluate . . . 55
5.2 Introdução aos Resultados . . . 55
5.2.1 Interpretação dos resultados obtidos . . . 56
5.2.2 Configuração Experimental . . . 58
5.2.3 Armazenamento . . . 59
5.2.4 Desempenho . . . 61

6 Conclusões e trabalho futuro **63**
6.1 Conclusões . . . 63
6.2 Objetivos Cumpridos . . . 64
6.3 Perspetiva de trabalho futuro . . . 66

I Apêndices **72**

A Detalhes dos resultados **73**
A.1 Explicação detalhada vantagens e desvantagens dos componentes do modelo híbrido . . . 73

B Listings **76**
B.1 Exemplificação de uma função teste . . . 76

Lista de Figuras

1 Exemplo de filtragem colaborativa 15
2 Exemplo de filtragem baseada em conteúdo 16
3 Exemplo de filtragem Híbrida 16

4 Representação de CRISP-DM 22
5 Processo mais adotado para projetos *data science* de acordo com [Saltz, 2022] 23
6 Representação da arquitetura do sistema de recomendações 27

7 Representação das idades dos clientes 39
8 Representação das idades dos clientes por faixa etária 39
9 Fragmentação da coluna address 40
10 Exemplo do modelo popularidade 45
11 Exemplo de filtragem Complementar 46
12 Número ideal de clusters 47

Lista de Tabelas

1 Planeamento da dissertação 7

2 Contrato de dados implementado dos produtos 34
3 Contrato de dados implementado dos clientes 35
4 Contrato de dados implementado das encomendas 36
5 *Dataframe* Produtos atualizado 42
6 *Dataframe* Clientes atualizado 42
7 *Dataframe* Encomendas atualizado 43
8 Vantagens, Desvantagens e Modelos Mitigantes no Modelo Híbrido 50

9 Resultados das métricas 56
10 Resultados da configuração experimental no modelo híbrido 58
11 Desempenho dos modelos com avaliação 62

Acrónimos

ACID Atomicidade, Consistência, Isolamento e Durabilidade.

API Application Programming Interface.

B2B Business-To-Business.

B2C Business-To-Consumer.

B2E Business-To-Employee.

CRISP-DM Cross-Industry Standard Process for Data Mining.

EDA Exploratory Data Analysis.

FBC Filtragem baseada no Conteúdo.

FC Filtragem Colaborativa.

IA Inteligência Artificial.

MAP Mean Average Precision.

ML Machine Learning.

NIST Instituto Nacional de Normas e Tecnologia.

RGPD Regulamento Geral de Proteção de Dados.

SEMMA Sample Explore Modify Model and Assess.

SR Sistemas de Recomendação.

SVD Singular Value Vecomposition.

TF-IDF Term Frequency-Inverse Document Frequency.

Capítulo 1

Introdução

1.1 Contexto

De acordo com Nills Nilson em [Nilsson, 1998], **IA** abrange a simulação de processos de inteligência humana por meio de sistemas computacionais. Esses procedimentos compreendem a capacidade de adquirir conhecimento, a aquisição de dados e regras, bem como o processo de raciocínio, o qual é o uso de regras preestabelecidas para chegar a conclusões provisórias ou conclusivas. Além disso, há também a aptidão para auto-retificação.

E-commerce define-se como a realização de transações envolvendo artigos ou serviços por via *Internet*, utilizando qualquer forma de comunicação eletrónica, inclusive transferências financeiras e os dados inerentes a esses processos, [Nanehkaran, 2013]. Por outras palavras, o comércio eletrónico refere-se à compra e venda de produtos ou serviços via uma plataforma *online*. Dentro das várias modalidades de *e-commerce* [Anev, 2021], destacam-se três: *business-to-business* (**B2B**), *business-to-consumer* (**B2C**) e *business-to-employee* (**B2E**). As modalidades mencionadas destacam-se das restantes por refletirem os tipos de transações que ocorrem numa plataforma *e-commerce*, estas cobrem as diversas necessidades e relações num ambiente de negócios digital.

O tipo de *e-commerce* **B2C** refere-se à venda de artigos ou serviços de uma loja virtual diretamente para o consumidor final, via plataforma *online*. A abordagem padrão a *e-commerce* **B2C** é a utilização de tecnologias digitais, entre as quais: plataformas de *e-commerce*, aplicações móveis ou mesmo meios de comunicação social para alcançar e envolver os consumidores. O objetivo é melhorar a experiência de compra do cliente, aumentar o conhecimento da marca e impulsionar as vendas, fornecendo uma forma conveniente, fácil e eficiente para os clientes adquirirem bens e serviços a qualquer momento. Esta dissertação terá foco no tipo de *e-commerce* **B2C**.

B2B aborda compra e venda de bens e serviços entre empresas e não o consumidor final, também por via plataforma *online*. Este método engloba desde fornecedores de matérias-primas até serviços de

logística e transporte. O objetivo resume-se em melhorar a eficiência e a relação custo-eficácia entre as empresas e criar oportunidades de negócio, mediante mercados *online* e outros canais digitais.

Por fim, **B2E** representa a utilização de plataformas e tecnologias *online* para facilitar possíveis transações e comunicação entre uma empresa e os seus funcionários. O objetivo de **B2E** é melhorar, internamente, a eficiência e produtividade de uma empresa. Este método promete simplificar processos internos e proporcionar aos colaboradores um acesso fácil às ferramentas e recursos necessários.

À medida que a sociedade evolui e exige cada vez mais modernização, ferramentas como o *e-commerce* estão em constante necessidade de expansão. Em adição, o número de plataformas *online* que oferecem os mesmos serviços aumentam a cada dia, provocando a necessidade de alcançar novas formas de se destacar entre a concorrência. Nesta dissertação, o *e-commerce* e **IA** vão-se relacionar na construção de um sistema de recomendação de produtos. Segundo Dr.Geoffrey Hinton, cientista de computação conhecido como o fundador da *deep learning*,[LeCun et al., 2015], sistemas de recomendação são sistemas de inteligência artificial que utilizam algoritmos para analisar e aprender padrões em grandes conjuntos de dados sobre os hábitos de consumo e comportamentos dos utilizadores. Estes padrões serão utilizados para fazer recomendações personalizadas para cada utilizador, baseadas nos seus interesses e preferências anteriores. Hinton defende que os sistemas de recomendação são importantes porque ajudam a tornar a vida do utilizador mais fácil, proporcionando-lhes recomendações precisas e relevantes sobre o que eles podem ou não gostar, ou precisar. O "pai do *deep learning*" afirma que esta é uma área em constante evolução, no qual a cada passo não só são derrubadas barreiras que outrora pareciam impossíveis, como também são estabelecidas novas barreiras relacionadas com eficiência e desempenho. Com a capacidade de processar e analisar dados de grande dimensão, os modelos de recomendação contribuem para as empresas adquirirem uma melhor compreensão sobre os seus clientes, antecipar as suas necessidades, e oferecer uma experiência de compra mais adaptada.

Esta dissertação será desenvolvida no âmbito proposto pela empresa Bsolus. A plataforma fornece soluções de comércio eletrónico **B2C**, **B2B** e **B2E**, a médias e grandes empresas, oferecendo uma plataforma digital para o *e-business* dos seus clientes. A Bsolus constrói e gere as plataformas de comércio eletrónico de empresas clientes, tal como a DeltaQ Portugal. A empresa oferece a gestão da loja *online* e respetiva lógica de negócios (clientes, produtos e gestão de encomendas), *marketing*, apoio, gestão de relacionamento e análise empresarial.

1.2 Motivação

A minha principal motivação de aceitar este projeto foi principalmente impulsionada pela possibilidade de estudar e integrar inteligência artificial num sistema destinado a uso comercial. A meu ver, a capacidade da **IA** para otimizar processos e analisar dados tem o potencial de revolucionar inúmeras áreas, o que torna **IA** um setor promissor.

A empresa Bsolus, especializada na criação e gestão de lojas de *e-commerce*, posiciona-se num mercado cada vez mais competitivo, onde a inovação é fundamental para se manter na vanguarda e conquistar a preferência do consumidor. Diante deste cenário desafiador, a empresa procura formas de se destacar e competir contra as inúmeras opções. A empresa reconheceu a relevância de adotar soluções inovadoras e resolveu que a **IA** seria um passo em direção ao futuro e uma oportunidade estratégica para aperfeiçoar os seus sistemas e serviços. Como primeiro passo nesta jornada, a integração de algoritmos **ML** apresenta diversas vantagens para os negócios *e-commerce*,[Kian, 2021].

A integração do modelo de recomendações proporciona uma personalização aprimorada na experiência do consumidor, oferecendo recomendações mais precisas e relevantes com base no comportamento de compra e preferências individuais. A relação entre recomendações e **IA** não só aumenta a satisfação do cliente, mas também fortalece a sua fidelização e aumenta as receitas da loja virtual. Como se não fosse motivo suficiente, a **IA** permite aumentar a capacidade de análise de dados em larga escala, permitindo à Bsolus compreender padrões de comportamento do consumidor e tendências de mercado de forma mais eficiente. Esta análise aprofundada contribui para a tomada de decisões estratégicas mais fundamentadas, melhorando a eficácia das campanhas de *marketing* e a seleção de produtos a serem destacados.

A procura por uma ferramenta agnóstica que integre **IA** com modelos de recomendação é motivada pela necessidade de flexibilidade e escalabilidade. A Bsolus reconhece que uma solução versátil, capaz de se adaptar a diferentes contextos empresariais, é mais valiosa, especialmente num mercado dinâmico e diversificado. A criação de uma ferramenta que requer intervenção mínima de programadores, é uma estratégia que visa otimizar recursos. Ao poupar tempo e dinheiro, a Bsolus poderá direcionar os seus esforços para áreas mais estratégicas, garantindo uma vantagem competitiva sustentável. Ademais, ao disponibilizar uma ferramenta de fácil uso para os negócios que a adotam, a Bsolus proporciona uma experiência simplificada, fazendo com que os utilizadores se sintam mais capacitados e atualizados relativamente às dinâmicas do mercado.

A motivação, detalhada no artigo [Kian, 2021], por trás da integração de **IA** em sistemas de reco-

mendação na Bsolus é impulsionada pela necessidade de se adaptar ao cenário competitivo, oferecer experiências personalizadas, analisar dados eficientemente, garantir flexibilidade e, ao mesmo tempo, otimizar recursos para se destacar como líder no mercado de *e-commerce*.

1.3 Objetivos

O presente projeto visa detalhar o desenvolvimento de um sistema de recomendações acoplado à plataforma Bsolus destinado melhorar a experiência dos utilizadores e aumentar o envolvimento dos clientes, o que, por sua vez, levará a um aumento nas vendas. Isto requer criar um sistema que consiga compreender e antecipar as preferências individuais de cada utilizador, fornecendo recomendações precisas e personalizadas.

Para a construção deste projeto, é fundamental estabelecer um contrato de dados que todas as plataformas envolvidas devem cumprir. Isto garante a consistência e a qualidade dos dados utilizados para treinar os algoritmos de aprendizagem automática, melhorando assim a precisão das recomendações. Além disso, o sistema deve ser modular, permitindo a sua adaptação a diferentes modelos de negócio e plataformas. Esta modularidade é essencial para assegurar a versatilidade do sistema e a sua aplicabilidade em qualquer empresa cliente.

Um requisito igualmente crucial é garantir a compatibilidade do sistema de desenvolvido com a estrutura existente da ferramenta Bsolus. O objetivo mencionado garante que o sistema de recomendações possa ser integrado com os restantes serviços da plataforma, criando uma solução operacional para os clientes da Bsolus.

Posto isto, os objetivos estabelecidos pela empresa responsável são:

1. Compreensão do fluxo de informação de uma plataforma de e-commerce;
2. Implementação de um método capaz de albergar qualquer plataforma associada à Bsolus e futuro treino dos dados;
3. Implementação de novas técnicas de **ML**, sendo estas as mais aceites e estudadas pela academia até o momento presente;
4. Análise dos resultados mediante métricas definidas pela equipa e processos de validação orientados a este tipo de problemas;
5. Garantir compatibilidade e comunicação com os restantes micros serviços da plataforma, mantendo sempre a estrutura estabelecida;

6. Implementação de um sistema de recomendações modular, capaz de abranger qualquer tipo de plataforma

1.4 Estrutura e Planeamento

1.4.1 Estrutura

O presente documento está organizado num conjunto limitado de capítulos, cada um dedicado a um tópico fundamental do projeto e a consequente análise detalhada dos desafios e possibilidades associadas ao desenvolvimento de um modelo de recomendação na plataforma Bsolus.

- **Introdução:** No primeiro capítulo, é dado o contexto sobre o tema de toda a dissertação, estabelecendo a motivação e os objetivos que guiarão o nosso projeto. Este capítulo irá definir as bases para todo o planeamento subsequente, delineando o que aspiramos alcançar e o que o sistema será capaz.

- **Estado de Arte:** No segundo capítulo, aprofundaremos a análise do contexto atual e o estado de arte de *e-commerce* e dos sistemas de recomendação, assim como temas relacionados. Este capítulo abrange os sistemas de recomendação propostos a desenvolvimento e prévia transformação de dados, casos de sucesso e possíveis obstáculos. Esta abordagem fornecerá informações valiosas e referências importantes para o desenvolvimento do nosso sistema na plataforma Bsolus.

- **Abordagem Proposta:** Neste capítulo será discutido todo o processo que não envolva diretamente a manipulação de dados ou a construção dos **SR**. Será apresentado o método de *data mining* escolhido, a lógica estabelecida e informações relevantes para compreender o projeto na totalidade.

- **Implementação:** Ao contrário do capítulo anterior, neste será discutido o processo que envolva diretamente a manipulação de dados e a construção dos algoritmos. Começaremos pela compreensão do negócio e dos dados, seguida pela preparação de dados, modelação, e, finalmente, a integração dos sistemas de recomendação. Este capítulo oferecerá uma visão completa do processo de implementação do projeto.

- **Resultados:** Exploraremos os resultados obtidos com os sistemas de recomendação, incluindo diferentes variações e o desempenho de cada modelo. Métricas de avaliação, como *precision* e *recall*, serão utilizadas para inferir a eficácia das recomendações. Além disso, discutiremos os

requisitos de armazenamento necessários para suportar os modelos em produção, assim como configurações experimentais nos modelos. O capítulo fornecerá uma visão abrangente dos resultados do projeto.

- **Conclusão e trabalho futuro:** Como conclusão deste projeto será resumido todo o trabalho realizado, destacando os principais objetivos alcançados. Este capítulo terá como destaque também possíveis direções futuras, como a incorporação de **B2B** e **B2E** nos **SR**, possíveis melhorias aos modelos de recomendação implementados e a incorporação de um assistente virtual acessível a todos os clientes. Este capítulo encerrará a dissertação, apontando para oportunidades de aprimoramento contínuo do sistema de recomendação desenvolvido.

1.4.2 Calendarização

A pesquisa e construção deste projeto foi realizada ao longo de um ano, com a iniciação do meu estágio profissional na empresa Bsolus, com data de início em outubro de 2022 e fim em outubro de 2023.Previamente à iniciação desta dissertação foi delineado um planeamento, no qual foi estabelecido um limite de tempo para cada tarefa principal deste projeto, demonstrado na tabela 1. Este calendário foi cumprido corretamente, dado que no mês de agosto de 2023 todos os objetivos definidos para este projeto estavam concluídos.

- **Material Introdutório**

  Nesta fase foi delineado o rascunho, o *background* e definições necessárias a saber para este projeto. Esta fase, constituída por **Introdução**, **Estado de arte** e **Problemas e desafios** foi essencialmente pesquisa, no qual o maior obstáculo foi entender quais os temas que deviam ser abordados no capítulo Estado de Arte. Posto isto, esta fase foi cumprida no limite estabelecido.

- **Arquitetura**

  Possivelmente a tarefa mais desafiante de todas, nesta fase resume-se todas as decisões tomadas ao logo desta dissertação não relacionadas diretamente com os modelos de recomendação. Esta fase, apesar também ter sido concluída nos limite estabelecidos, demorou mais do que esperado devido às inúmeras possibilidades em cada decisão, sendo cada uma delas devidamente estudada de modo a encontrar a mais eficiente possível.

- **Implementação dos modelos** Assim, tomadas as decisões necessárias, centrou-se na construção do modelo de recomendações, seguindo o método **CRISP-DM**. A esta fase foi dado, desde

início, um maior intervalo temporal para a sua execução, visto tratar-se não só do núcleo da dissertação como também do projeto em si e do estágio que decorria naquele momento.

- **Resultados e reavaliação**

  Com um limite de 3 meses, nesta fase avaliou-se os resultados obtidos dos modelos. Uma vez que inicialmente não foram obtidos os resultados desejados, avaliou-se de novo desde a construção de dados até à construção dos algoritmos **ML**. A esta fase também foi atribuída uma parcela considerável nesta calendarização com o propósito de aperfeiçoar os modelos da melhor forma possível.

- **Documentação:** Com a construção do modelo de recomendações concluída e propriamente revista, passou-se à documentação deste projeto.

Tarefa	Out	Nov	Dez	Jan	Fev	Mar	Abr	Mai	Jun	Jul	Ag	Set	Out
Material Introdutório	●	●	●	●									
Arquitetura					●	●	●						
Implementação modelos						●	●	●	●				
Resultados e reavaliação									●	●	●		
Documentação									●	●	●	●	●

Tabela 1: Planeamento da dissertação

Capítulo 2

Estado da arte

2.1 O comércio

2.1.1 De comércio a e-commerce

O cenário atual do *e-commerce*, ao qual a sociedade tem acesso, não surgiu linearmente. Pelo contrário, este fenómeno foi fruto de um processo longo e meticuloso que teve o seu início na década de noventa, com a introdução da "*internet* comercial" e, posteriormente, o surgimento dos primeiros catálogos *online*. Os catálogos em questão permitiam aos utilizadores visualizar os produtos e encomendar os mesmos via telefone ou *fax*, [Tian and Stewart, 2008].

À medida que este setor evoluía e a *internet* conquistava a sociedade, as lojas virtuais começaram a investir em novas formas de atrair mais clientes e simplificar o processo de compra. O aprimoramento da experiência do consumidor tornou-se uma prioridade para estes pioneiros de *e-commerce*. A evolução da *internet* produziu um fenómeno simbiótico: à medida que as lojas virtuais evoluíam, o interesse e a interação dos consumidores aumentava significativamente. Não somente este ciclo foi benéfico para o *e-commerce*, como também demonstra a intrínseca ligação entre o progresso tecnológico e a crescente procura por uma forma mais conveniente e acessível de realizar transações comerciais.

Como qualquer área que exige evolução, o *e-commerce* deu um grande passo em direção ao futuro quando incorporou **IA** nas suas plataformas. Esta junção revolucionou para sempre como as empresas interagem com os clientes e gerem as suas operações. Desde a implementação de sistemas de recomendações personalizadas, tema que em que se centra esta dissertação, à utilização de assistentes virtuais, *chatbots* e até mesmo mecanismos de deteção de fraudes, [Dornadula and Geetha, 2019] estas ferramentas impulsionadas por **IA** oferecem suporte e aumentam a probabilidade de vendas adicionais, melhorando a experiência do cliente, tornando-a mais relevante e atrativa, [Kian, 2021]. Atualmente, o *e-commerce* é uma peça fundamental da economia global e não se espera um abrandamento no seu

crescimento.

Como se o constante crescimento desta nova ferramenta não fosse suficiente, a pandemia COVID-19 acelerou a popularidade desta ferramenta. Devido às restrições de deslocamento e às medidas de distanciamento social, a sociedade procurou uma alternativa às lojas físicas. Isto levou a um aumento significativo nas vendas e no tráfego do *e-commerce*. Produtos que outrora tinham pouca procura digital, como alimentação, artigos de limpeza ou equipamentos de proteção, tiveram um aumento descomunal nas suas vendas. Empresas que não tinham presença *online* foram forçadas a estabelecer presença rapidamente para se manterem competitivas com a concorrência. O crescimento do comércio eletrónico foi tão grande que as companhias enfrentaram dificuldades operacionais, como a escassez de mão de obra, atrasos nos fornecedores e dificuldades para atender à crescente demanda. Infelizmente, o aumento do *e-commerce* também levou a um aumento no comércio de fraudes. Em geral, o *e-commerce* desempenhou um papel crítico na manutenção da continuidade dos negócios e na garantia de acesso a bens de consumo necessários durante a pandemia COVID-19, [Susmitha, 2021].

2.1.2 Relação entre e-commerce, plataformas e negócio-cliente

"O uso de tecnologias de informação e comunicação para comprar e vender bens e serviços *online*", foi assim que **NIST**, agência governamental dos Estados Unidos responsável por estabelecer padrões e metodologias para a pesquisa e desenvolvimento de tecnologias de informação e comunicação, em 1997, estabeleceu a primeira definição oficial. Desde então, esta definição foi alargada e aperfeiçoada para incluir outros aspetos do *e-commerce*, como a comercialização, a publicidade e a entrega dos respetivos produtos. Em 2021, a definição de *e-commerce* foi atualizada para "A utilização de comunicações eletrónicas e tecnologia de processamento de informação digital nas transações comerciais para apoiar e melhorar a compra e venda de bens e serviços".

De acordo com [Tian and Stewart, 2008], *e-commerce* refere-se à execução de transações comerciais através da *internet*. Tais transações abrangem uma variedade de atividades, desde a compra e venda de bens tangíveis até à prestação de um serviço *online*. Quanto às plataformas digitais, estas são os espaços virtuais, geridos por empresas reais, a partir dos quais os consumidores podem aceder e realizar as referidas transações. Estas plataformas podem assumir a forma de *websites*, aplicações móveis ou até contas em redes sociais, o necessário para oferecer uma experiência de compra conveniente e acessível aos seus clientes.

No contexto **B2C**, o foco principal recai na relação direta entre a empresa e o consumidor final. Neste tipo de *e-commerce*, o objetivo das empresas é criar estratégias para atrair e envolver o seu público-alvo,

focando-se não só na venda de produtos, mas também na construção de uma posição de confiança entre o cliente e a plataforma. A personalização, a experiência do utilizador e a eficiência nas transações são componentes cruciais nesta equação.

Para tal, as empresas esforçam-se para entender as necessidades e preferências dos consumidores individuais, enquadrando a criação de campanhas de *marketing* direcionadas à otimização da experiência do utilizador e a implementação de soluções tecnológicas que facilitem o fluxo de compra. Nesta construção, a personalização desempenha um papel fundamental, respondendo à expetativa dos consumidores de receber ofertas e interações adaptadas às suas preferências e histórico de compras. A confiança é um dos principais pilares na estratégia **B2C**, uma vez que os consumidores estão dispostos a partilhar informação pessoal e financeira apenas quando sentem que a empresa é de confiança. Práticas transparentes, a garantia de uma segurança robusta dos dados e um serviço de apoio ao cliente tornam-se cruciais para a concretização da estratégia.

2.2 Integração dos sistemas de recomendação

2.2.1 Sistemas de Recomendação

Antes de se aprofundar na complexidade das recomendações em ambientes de comércio eletrónico, é necessário compreender a definição de sistema de recomendação. Sistemas de recomendação são algoritmos "inteligentes" que, ao analisar os padrões de comportamento do utilizador, o seu histórico de compras e preferências pessoais, fornecem sugestões personalizadas ao cliente. Na sua essência, os sistemas de recomendação procuram melhorar a experiência do utilizador, tornando-a mais relevante, eficiente e, consequentemente, mais agradável. Assim que são integrados, os modelos de recomendação fazem parte da experiência do cliente na plataforma.

É uma ferramenta que tende a operar em "tempo real", analisando constantemente os dados mais recentes para aprimorar a precisão das suas recomendações. Neste contexto, um modelo de recomendação torna-se essencial para proporcionar aos consumidores uma experiência de compra única e adaptada às suas preferências, necessidades e interesses específicos.

Segundo o artigo [Alamdari et al., 2020], a emergência dos sistemas de recomendação remonta à vastidão de opções disponíveis numa plataforma *online*. Perante uma plataforma que disponha um conjunto considerável de produtos, o cliente é sobrecarregado com informação e muitas vezes não consegue encontrar o produto que procura. No entanto, com a utilização de um modelo de recomendação, toda esta informação será filtrada e apresenta-se apenas produtos relevantes para o cliente. Não só melhora a

experiência do cliente como também impulsiona a venda de produtos que se encontram escondidos no *website*.

2.2.2 Integração de Inteligência Artificial

A incorporação de **IA** nos sistemas de recomendação representa um marco significativo na evolução das plataformas, conferindo-lhes uma capacidade de adaptação e personalização sem precedentes. A introdução da **IA** nos sistemas de recomendação remonta às últimas décadas do século XX, quando as capacidades computacionais começaram a permitir a implementação de algoritmos mais complexos. À medida que o poder de processamento aumentou e os algoritmos foram refinados, a **IA** assumiu um papel central na revolução destes sistemas. A inteligência artificial tornou-se não apenas uma adição promissora, mas o futuro dos sistemas de recomendação. O reconhecimento da capacidade da **IA** para lidar com conjuntos consideráveis de dados, identificar padrões complexos e aprender com os seus erros foi o catalisador para revolucionar esta industria, [Zhang et al., 2021].

A introdução da **IA** introduziu uma série de vantagens aos sistemas de recomendação e, por extensão, às plataformas comerciais aderentes. Primeiramente, a capacidade de processar grandes volumes de dados em tempo real possibilitou uma personalização mais personalizada, aumentando a relevância das recomendações. Em adição, os algoritmos de **ML** permitiram a adaptação dinâmica às mudanças nas preferências dos utilizadores, melhorando a precisão ao longo do tempo. Teoricamente, algoritmos **ML** conseguem aprender e ajustar-se continuamente com base no *feedback* do utilizador. Um atributo igualmente vantajoso é a capacidade de explorar possíveis relações entre diferentes artigos, proporcionando sugestões mais diversificadas e, por conseguinte, enriquece a experiência do utilizador.

Num mundo onde a sobrecarga de informação é a norma, os sistemas de recomendação desempenham um papel crucial na simplificação da tomada de decisões do utilizador. A sua evolução, desde as abordagens iniciais até à integração da **IA**, reflete a constante busca por sistemas mais inteligentes e adaptáveis. Esta interseção entre a evolução histórica e a atual integração de tecnologias avançadas destaca não apenas o progresso técnico, mas também a relevância destes sistemas na moldagem da era digital moderna, [Lü et al., 2012].

2.2.3 O Caso de Sucesso

Tal como já referenciado anteriormente, as plataformas *online e-commerce* adotaram **SR** como uma forma de melhorar a experiência do cliente e aumentar os lucros. Sem termos noção, estes modelos de recomendação estão integrados com o nosso quotidiano de interações digitais, estando presentes em

reconhecidas plataformas como *Facebook*, *Netflix*, *Spotify* ou *YouTube*.

No entanto, alguém teve que ser o pioneiro na investigação e implementação de **SR** e, consequentemente, ser a combustão e o incentivo para o resto das empresas seguirem os seus passos, esta empresa foi a **Amazon**. A *Amazon* iniciou o percurso com o lançamento desta nova tecnologia em 1996. Já nesta altura, o modelo consistia num sistema de **FC** que recomendava produtos aos clientes com base nas suas compras anteriores e nas compras de outros clientes que adquirissem produtos semelhantes, [Jiang et al., 2019].

Em 2003,[Smith and Linden, 2017], o algoritmo foi lançado a uma escala global através da plataforma da *Amazon* e qualquer página da plataforma incluía algum tipo de recomendação. Sugestões de produtos com base nas compras anteriores encontravam-se na página inicial e nos resultados de procuras, na página de registo de completar a compra exibiam produtos semelhantes aos que o utilizador lá teria posto, impulsionando o cliente a adicionar mais produtos à sua encomenda no momento antes do pagamento. Por fim, a plataforma permitia que os clientes avaliassem os produtos e deixassem comentários, o que permitia ao sistema aprender e adaptar-se às preferências dos clientes. Esta estratégia continua a ser eficaz e utilizada por inúmeras plataformas *e-commerce*.

Atualmente, de acordo com [Lü et al., 2012],na Amazon, entre 20% e 40% das vendas devem-se a produtos que não pertencem aos 100 000 produtos mais vendidos da loja. Um sistema de recomendação pode, portanto, ter um impacto significativo nas vendas de uma empresa. Vários estudos foram realizados para comprovar a veracidade destes modelos e se realmente têm um impacto positivo no e-commerce. Entre eles estão: Em 2003, um estudo publicado na revista "*Management Science*" descobriu que o sistema de recomendação da *Amazon* aumentou as suas vendas em 29%. Este estudo analisou os dados do comportamento do cliente e vendas da *Amazon* entre 1995 e 2003. A pesquisa demonstrou que a utilização deste sistema foi uma das principais razões para o crescimento das vendas nesse período; em 2006, um estudo publicado na revista "*ACM Transactions on Internet Technology*" que descobriu que o sistema de recomendação da Amazon aumentou as vendas de livros recomendados em 153%; um relatório do *Forrester Research* de 2017 descobriu que as empresas que usavam **SR**, como o da Amazon, tinham melhor desempenho de negócios em comparação com aquelas que não os usavam. O relatório apontou que as empresas que implementam **SR** veem aumento em conversões, vendas, e também melhoria na satisfação do cliente. Isso mostra a importância dos **SR** para o sucesso do negócio e-commerce. Estes são apenas uma fração dos inúmeros estudos, análises e relatórios desenvolvidos que comprovam a relevância dos modelos de recomendação no *e-commerce*.

2.3 Recomendações e Tratamento de Dados

No contexto de *e-commerce*, um sistema de recomendação é uma ferramenta que visa sugerir produtos relevantes aos consumidores com base nas suas preferências e/ou comportamentos de compra. Este tipo de sistemas recolhem dados sobre os utilizadores e os seus interesses e, a partir de algoritmos **ML** para identificar padrões, "promete" fazer recomendações com as quais o cliente se relacione. Estas recomendações, por exemplo, podem e devem incluir produtos menos populares que possam estar "escondidos", mas relevantes para o cliente em questão, uma vantagem para as duas extremidades da conversa.

2.3.1 A Importância do Tratamento de Dados

O sucesso de um sistema de recomendação em plataformas *e-commerce* não se trata apenas da mera criação de algoritmos **ML**, mas a junção destes com um tratamento adequado aos dados provenientes das plataformas, [Wongsuphasawat et al., 2019].

Em 1977, Jonh W. Tukey, [Tukey et al., 1977], introduziu e realçou a importância da análise exploratória de dados (**EDA**) como uma abordagem fundamental na compreensão dos dados antes de realizar qualquer análise formal. Em termos ligeiros, Tukey defendeu que qualquer analista de dados deveria primeiro visualizar e explorar os dados graficamente para obter informações preliminares sobre padrões, tendências e características distintas antes de prosseguir com análises mais complexas. No seu livro, afirma que a abordagem a seguir não é conjunto de técnicas, mas sim uma "filosofia" sobre como o tratamento de dados deve ser analisado. Segundo o seu livro [Tukey et al., 1977], esta abordagem é caracterizada por:

1. **Uso de Gráficos e Visualizações:** Salienta a importância de representar os dados visualmente antes de se tomar qualquer decisão. Entre inúmeras opções, tais como histogramas, gráficos de caixa e gráficos de dispersão, estes destacam-se como ferramentas poderosas para explorar a distribuição e a estrutura dos dados fornecidos.

2. **Descoberta de Padrões e Tendências:** A **EDA** aspira identificar padrões, tendências e características relevantes nos dados. Esta descoberta inicia-se por meio de técnicas gráficas e descritivas que revelam informações importantes sobre o conjunto de dados.

3. **Flexibilidade e Abertura para Descobertas:** A filosofia **EDA** encoraja os analistas a manterem uma mente aberta e ajustarem as suas conclusões com base nas informações descobertas durante todo o processo de exploração.

Através da **EDA**, é possível descobrir informações cruciais que estão ocultas nos dados, garantindo que os algoritmos de recomendação baseiam-se no máximo de informações possíveis, sendo estas sólidas e fiáveis. Este método, ao educar os sistemas de recomendação a compreender melhor o comportamento do cliente, origina recomendações mais precisas e pertinentes. Deste modo, ao combinar algoritmos de **ML** com uma análise cuidadosa e interpretação dos dados, as plataformas de comércio eletrónico alcançam resultados superiores em termos de lucro e satisfação dos utilizadores.

2.3.2 Filtragem Colaborativa

A abordagem Filtragem Colaborativa (**FC**), figura 1, define-se como a procura de semelhanças entre clientes, consoante os seus comportamentos, ações e preferências. Sem fugir à regra, se um grupo de utilizadores têm uma opinião sobre um produto, um cliente com características semelhantes terá a mesma opinião sobre o mesmo produto, [Fayyaz et al., 2020].

Posto isto, existem dois tipos principais de **FC**:

- **Baseada no utilizador:** As recomendações são feitas com base nas semelhanças entre os perfis dos utilizadores, isto é, cujo comportamento em avaliações de produtos se assemelhe ao utilizador a quem se destina as recomendações, [M. Hasan and Ahmed, 2016]. Exemplificando, se um utilizador "A" avaliou positivamente os mesmos filmes que outros utilizadores também avaliaram positivamente, o sistema recomenda filmes que esses utilizadores gostaram e que o utilizador A ainda não viu;

- **Baseada no artigo:** As recomendações são baseadas nas características dos produtos, isto é, procura similaridades entre avaliações e características dos produtos, [B. Sarwar and Riedl, 2001]. Exemplificando, se um utilizador "A" viu vários filmes sobre uma categoria "X", o sistema recomenda outros filmes relacionados à mesma categoria que foram bem recebidos por outros utilizadores com interesses semelhantes.

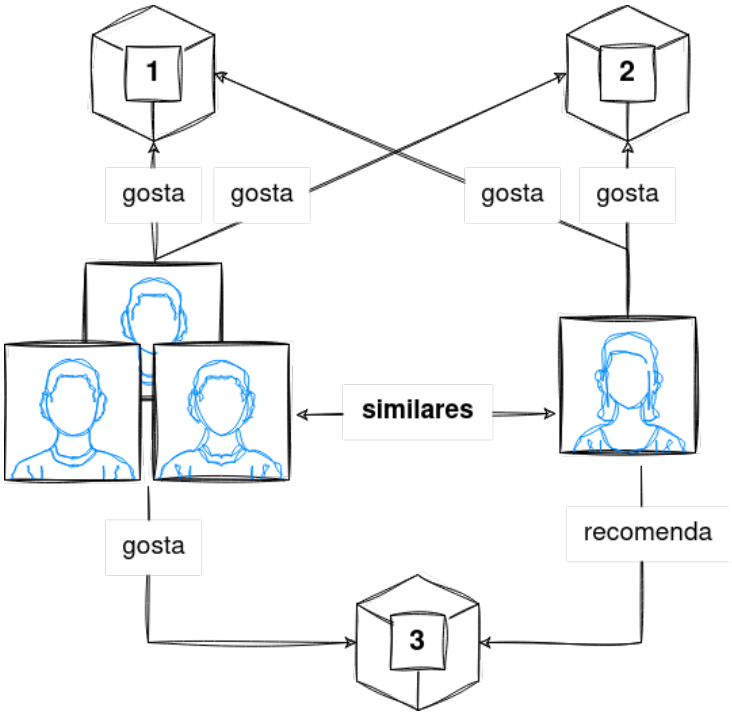


Figura 1: Exemplo de filtragem colaborativa

2.3.3 Filtragem Baseada no Conteúdo

Enquanto a **FC** se baseia nas interações e características entre os utilizadores, a filtragem baseada em conteúdo (**FBC**), como podemos inferir na figura 2, baseia-se nas semelhanças entre os artigos para fazer recomendações. Rege-se pela regra que se um utilizador gostar de um determinado produto, gostará também de produtos com características semelhantes, [Fayyaz et al., 2020].

Em **FBC**, analisam-se os atributos dos produtos com os quais um utilizador interagiu anteriormente (por exemplo, visto, comprado, ou classificado) e utiliza esta informação para recomendar os produtos com mais semelhanças. Exemplificando, se um utilizador gostar de muitos filmes de uma determinada categoria, conjuntos de atores e/ou diretor, é provável que o algoritmo lhe recomende filmes que se enquadrem nessas características.

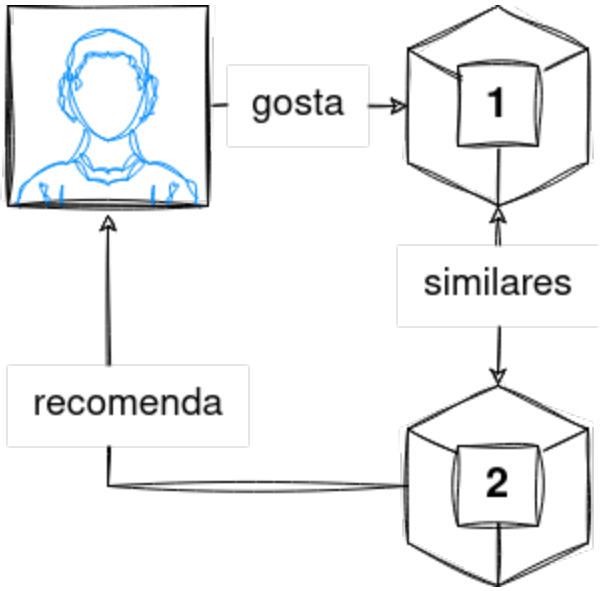


Figura 2: Exemplo de filtragem baseada em conteúdo

2.3.4 Filtragem Híbrida

Um modelo de recomendação híbrido, figura 3, consiste na união de dois ou mais modelos de recomendação individuais. Ao albergar mais que um modelo de recomendação, isto permitirá fornecer recomendações mais precisas e eficazes aos utilizadores. A combinação de modelos permite mitigar desvantagens que os modelos possam ter com as vantagens que outro modelo proporciona.

Esta estratégia permite superar as limitações individuais de cada método e oferecer recomendações mais personalizadas e diversificadas. Em situações de recomendação de produtos, as duas técnicas padrão são: **FC** e **FBC**. Ao integrar a **FC**, que tem em conta o comportamento de clientes semelhantes, com a **FBC**, que analisa as características dos produtos, o sistema pode compensar as limitações de escalabilidade e disparidade dos dados, ao mesmo tempo que considera as preferências individuais dos utilizadores. Para a estratégia de combinação dos componentes do modelo híbrido existem várias possibilidades, [Fayyaz et al., 2020] entre as quais: *weighted*, *switching*, *mixed*, *feature combination*, *cascade*, *feature augmentation*.

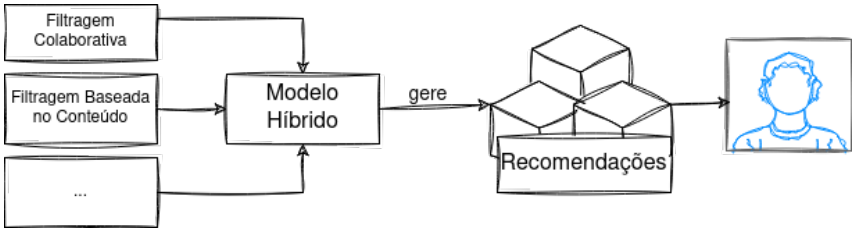


Figura 3: Exemplo de filtragem Híbrida

2.4 Problemas e desafios

A implementação de um sistema de recomendações em plataformas *e-commerce* é um desafio complexo. Envolve lidar com um volume significativo de dados, questões de privacidade, desenvolver algoritmos precisos e otimizar constantemente o sistema. Mesmo após décadas de progressão, alguns obstáculos permanecem desde 1990, quando foram inicialmente introduzidos. Neste capítulo serão os problemas que um **SR** genérico enfrenta e possíveis abordagens pré-definidas para os reduzir o máximo possível.

2.4.1 Análise de dados

Na criação de um serviço de recomendações autónomo um dos primeiros desafios é o tratamento dos dados recolhidos. Ao contrário de um projeto tradicional, em que o tratamento é feito especificamente para o projeto em questão, o serviço desenvolvido terá de conseguir lidar com informações de diversas plataformas. Independentemente da origem dos dados, o sistema deve processá-los e fornecer recomendações aos clientes da plataforma em questão. A solução mais apeladora para este desafio é a implementação de um conjunto de normas que todas as plataformas envolvidas devem cumprir. Este contrato estabeleceria regras para a formatação e disponibilização dos dados, garantindo que possam ser compreendidos e processados pelo sistema de recomendações autónomo.

2.4.2 Cold-start

Segundo [Sharma and Gera, 2013], o problema *cold-start* refere-se à dificuldade em formular recomendações precisas para novos utilizadores ou novos artigos que ainda não tenham sido classificados e/ou revistos. Este problema subdivide-se em três: o problema do novo utilizador, o problema do novo artigo e o problema do novo sistema. No caso de um novo utilizador, há pouca informação disponível sobre o mesmo, e, para um novo produto, geralmente não há classificações disponíveis.

No caso de um novo utilizador, que adere a uma plataforma de *e-commerce*, este não possui histórico de compras anteriores, nem classificações, nem revisões. Sem esta informação, o algoritmo designado não pode fazer qualquer tipo de recomendação ao utilizador. Uma forma eficaz de contrariar este problema será solicitar ao novo utilizador informações demográficas (idade, género, localização, interesses) sobre o mesmo de modo a encontrar semelhanças a outros utilizadores. Da mesma forma, no caso de um novo produto, que acabou de ser adicionado a uma plataforma digital, não tem classificações ou revisões, consequentemente o sistema de recomendações não tem como fornecer o produto como recomendação ao utilizador (problema do novo produto). De modo a introduzir este produto na lista de recomendações,

deve-se munir o mesmo do máximo de informação possível e semelhanças para com outros produtos para que este seja introduzido na **FBC**.

Este problema afeta principalmente a **FC**, que não consegue fazer recomendações úteis tanto para novos produtos quanto para novos utilizadores. No entanto, algoritmos baseados em conteúdo, tal como **FBC**, podem fornecer recomendações no caso de novos produtos, pois não dependem de informações de classificação anteriores de outros utilizadores para recomendar o produto.

2.4.3 Escalabilidade

Recorrendo ao mesmo artigo, [Sharma and Gera, 2013], escalabilidade é a capacidade de um sistema de lidar com um conjunto exponencial de dados sem interromper o fluxo do sistema. No contexto delineado, o problema de escalabilidade refere-se à dificuldade de manter o desempenho e a eficiência dos algoritmos, caso o número de informações aumente exponencialmente. À medida que a base de dados aumenta, a complexidade computacional dos algoritmos de recomendação também aumenta, o que pode levar a problemas de escalabilidade.

Dado um exemplo real, o aumento do número de clientes aderentes à plataforma e, posteriormente, clientes a que serão disponibilizados recomendações, torna o algoritmo **FC** mais complexo e computacionalmente dispendioso, uma vez que este tem em conta a semelhança entre os clientes para gerar recomendações. Da mesma forma, um algoritmo **FBC**, no qual as recomendações são baseadas na similaridade dos artigos, pode se tornar computacionalmente custoso à medida que o número de produtos no sistema aumenta.

Uma solução para diminuir o problema de escalabilidade consiste em utilizar técnicas de amostra, em que é selecionado aleatoriamente um subconjunto de dados. Esta solução diminui substancialmente a complexidade computacional e aumenta o desempenho, mas pode significar na redução de recomendações precisas.

2.4.4 Dispersão

De acordo com [Sharma and Gera, 2013], a dispersão de dados tem uma grande influência na qualidade das recomendações. Regra geral, as avaliações e comentários representam-se em forma de matriz cliente-produto e, uma vez que a maioria dos clientes apenas avalia uma pequena fração dos produtos, esta matriz fica com bastantes valores em falta, resultando em avaliações esparsas. Como se não bastasse, com o aumento do número de utilizadores e produtos, as dimensões da matriz aumentam e a dispersão se torna mais evidente.

Por exemplo, se um cliente tiver classificado apenas um pequeno número de artigos, o sistema de recomendação tem informação limitada sobre as suas preferências e comportamento, o que pode levar a recomendações imprecisas. Da mesma forma, se um artigo só foi classificado por um pequeno número de utilizadores, o sistema de recomendação tem informação limitada sobre a sua popularidade e relevância, o que também pode levar a recomendações imprecisas.

2.4.5 Super especialização

O problema de recomendar produtos a clientes tendo em conta somente as suas preferências e características é que essas recomendações são limitadas à variedade entre a semelhança entre os produtos e pode impedir o cliente de conhecer produtos novos, não relacionados com as suas prévias compras, [Sharma and Gera, 2013]. Este obstáculo impede o utilizador de descobrir novos artigos e outras opções disponíveis. No entanto, a diversidade de recomendações é uma característica desejável em todos os sistemas de recomendação. Uma forma de combater este problema é, por exemplo, recomendar os produtos mais populares na plataforma, não tendo as características individuais do cliente mas sim considerando todos os clientes como um só.

2.4.6 Outros

SR não sofrem apenas de problemas de infraestrutura, mas também desafios que são apresentados à esmagadora maioria da tecnologia utilizada no nosso quotidiano, apesar de não ser o foco desta dissertação, estes são detalhadamente discutidos em [Di Noia et al., 2022]

- **Privacidade e segurança:** Sistemas dependem frequentemente dos dados dos utilizadores para fazer recomendações. A recolha de dados pessoais para melhorar as recomendações deve ser equilibrada com a proteção dos dados e o cumprimento das regulamentações de privacidade, estabelecida pela **RGPD**.

- **Enviesamento e imparcialidade:** Apesar de inúmeros esforços, em processos que envolva uma quantidade considerável de dados, estes podem perpetuar o preconceito e a discriminação se a informação para formar o sistema forem tendenciosos ou se os algoritmos utilizados para fazer as recomendações não forem justos.

- **Falta de transparência:** Algoritmos **ML**, para o cidadão comum, podem ser complicados de compreender e interpretar, o que pode levar a uma falta de confiança e adoção.

Capítulo 3

Abordagem Proposta

A construção de um serviço de recomendação vai muito além da "simples" criação dos modelos de recomendação. Questões como onde serão armazenados os resultados ou até como será feita a comunicação entre o serviço de recomendações e o resto da plataforma são questões que necessitam de resposta muito antes de inicializar os algoritmos **ML**. A discussão destes temas paralelos à construção dos modelos reconhece a complexidade e a amplitude deste desafio.

A construção e desempenho do modelo de recomendação é apenas uma peça do *puzzle*. A infraestrutura que os sustenta, a capacidade de integração harmoniosa com a restante plataforma e a escolha ponderada das tecnologias a utilizar desempenham um papel essencial no desfecho final. As decisões de estrutura requerem um planeamento aprofundado, dado que podem ter um impacto direto na experiência do cliente, na escalabilidade do sistema e, por fim, no sucesso das estratégias de recomendação personalizadas.

Este capítulo fornece uma visão aprofundada de como abordar estas questões fundamentais, que devem ser abordadas antes mesmo de iniciarmos o tratamento dos dados e a implementação dos algoritmos de recomendação.

3.1 Metodologia de Mineração de Dados

Segundo [Han et al., 2022], *Data mining* é o processo de identificar padrões interessantes e obter conhecimento a partir de uma abundância de dados. O processo referido é fundamental em campos como análise de dados, **ML** e **SR**, pois permite transformar dados brutos em conhecimento útil. Em qualquer projeto de grande dimensão, caracterizado por volumes significativos de informação, é fundamental seguir um método estruturado de trabalho. Seguir uma metodologia não só minimiza contratempos como maximiza a eficácia na extração de informação e diminui o tempo investido e recursos necessários. Posto isto, tendo em conta o projeto estabelecido, a escolha ficou entre duas metodologias: **CRISP-DM** e **SEMMA**.

A abordagem **SEMMA** é uma metodologia proposta pela *SAS Institute*, focada em processos de mineração de dados, constituída por:

1. **Sample (Amostra):** Seleção de dados que representem o conjunto de dados escolhido.
2. **Explore (Exploração):** Análise exploratória dos dados para compreensão inicial.
3. **Modify (Modificação):** Pré-processamento e transformação do conjunto de dados escolhido.
4. **Model (Modelação):** Aplicação de técnicas de modelação para desenvolver o modelo preditivo.
5. **Assess (Avaliação):** Avaliação da eficácia do modelo e, caso seja necessário, ajuste no modelo estabelecido para cumprir os objetivos.

O metodologia **CRISP-DM**, representado pelo figura 4, é um conjunto de seis etapas interligadas, [Schröer et al., 2021]:

1. **Compreensão do Negócio:** Identificar as necessidades das partes interessadas e estabelecer objetivos realistas. Estabelecer as bases para o projeto.
2. **Compreensão dos Dados:** Compreender os dados disponíveis para o projeto, isto é, recolher, explorar, avaliar e identificar problemas relativamente à qualidade dos dados, garantindo que estejam prontos para análise.
3. **Preparação dos Dados:** Identificar variáveis relevantes, a transformação dos dados (normalização, codificação de variáveis categóricas) e a divisão entre dados de treino e teste.
4. **Modelação:** Desenvolver os modelos de recomendação selecionados com base na informação processada.
5. **Avaliação:** A partir de métricas de desempenho previamente estabelecidas, avaliar a qualidade e eficácia dos modelos desenvolvidos. Caso estes não cumpram os objetivos estabelecidos, a metodologia dita que se volte ao passo compreensão de negócios.
6. **Lançamento:** Implementar os modelos resultantes num ambiente de produção capaz de ser utilizado no mundo real.

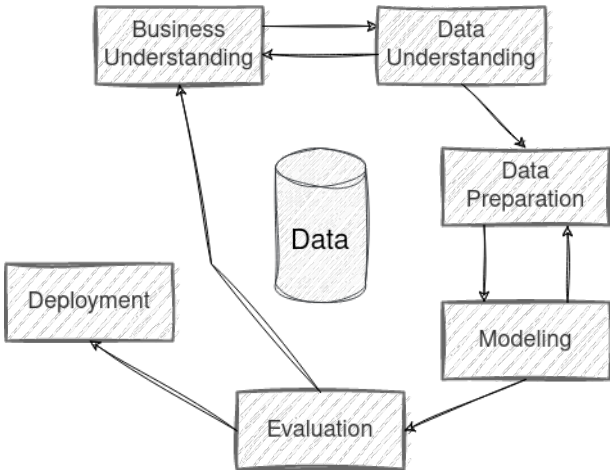


Figura 4: Representação de CRISP-DM

Embora ambas as abordagens sigam uma lógica sequencial, o **CRISP-DM** destaca-se por abranger todos os passos necessários para a construção de um sistema de recomendação *e-commerce* de grande envergadura.

Ao incorporar etapas como "Compreensão de negócio" e "Lançamento", esta abordagem incorpora melhor a perspetiva de negócios no processo de desenvolvimento, o que é crucial num projeto destinado a uso comercial. Enquanto o **SEMMA** foca-se mais na modelação estatística, o modelo escolhido equilibra a análise estatística com uma sólida compreensão dos objetivos comerciais.

Reforça-se que a metodologia **CRISP-DM** fornece um guia detalhado para cada fase do processo de mineração de dados, especialmente adaptado a projetos que tendem a ser utilizados no mundo real. As seis etapas, definidas pela metodologia, cobrem o projeto de início a fim, desde o estabelecimento de objetivos ao lançamento do serviço construído. Um benefício crucial do **CRISP-DM** é a constante avaliação dos sistemas de recomendação desenvolvidos. À medida que o projeto avança, é fundamental monitorizar e aperfeiçoar constantemente os algoritmos de forma gerar recomendações cada vez mais precisas. A constante avaliação facilita uma abordagem iterativa, garantindo que o sistema de recomendação é refinado à medida que novas informações e desafios surgem.

Posto isto, para a implementação de sistemas de recomendação inteligentes em plataformas de comércio eletrónico, a **escolha do CRISP-DM** destaca-se por não só considerar a perspetiva técnica, mas também o aspeto comercial e a monitorização contínua.

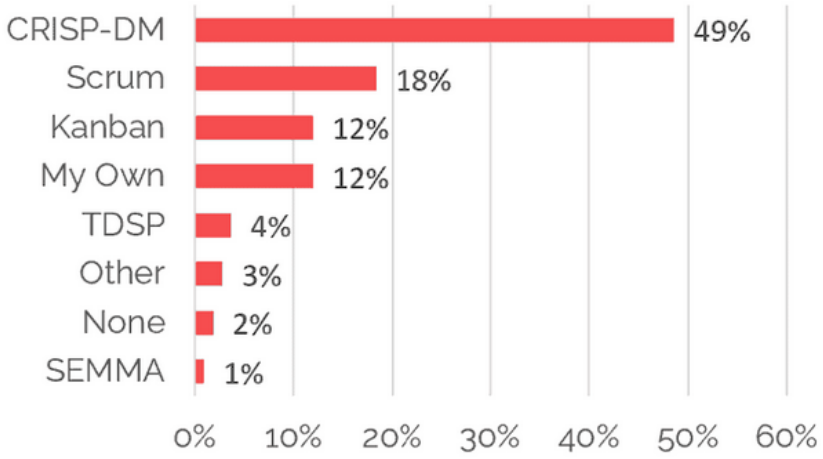


Figura 5: Processo mais adotado para projetos *data science* de acordo com [Saltz, 2022]

Segundo [Saltz, 2022], no ano de 2022, o modelo **CRISP-DM** continua a ser o processo mais utilizado e mais pesquisado.

3.2 Ferramentas de programação

3.2.1 Linguagem de programação

A escolha da linguagem de programação é um passo crucial na construção de qualquer microserviço. Apesar de a linguagem para a construção do modelo de recomendações tenha sido selecionada previamente ao meu envolvimento, pelos responsáveis do projeto, a linguagem *python* revelou-se como a melhor opção entre as linguagens mais populares para a implementação de um processo **ML**.

- *Python* é conhecida pela sua sintaxe simples e fácil compreensão, tornando-a adequada para programadores de todos os níveis de experiência.
- Relativamente a tarefas de aprendizagem, análise e processamento de dados, *python* oferece uma vasta gama de bibliotecas, tais como *Scikit-learn*, *TensorFlow* e *PyTorch*, que facilitam a construção e implementação de algoritmos de **ML**
- Ao ser uma linguagem atual e altamente utilizada, a linguagem possui uma comunidade de programadores ativos, provocando uma abundância de recursos disponíveis, desde documentação até *fóruns* de suporte, extremamente úteis no desenvolvimento e resolução de problemas.

- Oferece diversas ferramentas e bibliotecas para processamento eficiente de grandes conjuntos de dados, tornando-o ideal para lidar com os volumes de dados frequentemente encontrados em sistemas de recomendação de produtos.

- Ao ser uma das linguagens mais populares para desenvolvimento de inteligência artificial e **ML**, significa haver uma abundância de recursos disponíveis e uma ampla adoção na indústria.

3.2.2 Ferramentas de programação

A ferramenta PostgreSQL foi a escolhida para o armazenamento de dados. Apesar de o MongoDB ser uma ferramenta similar, PostgreSQL foi a decisão final pelas seguintes razões:

- Oferece um modelo de dados consistente e robusto, permitindo a definição precisa de esquemas e relações entre as entidades armazenadas. Adicionalmente, o PostgreSQL suporta transações **ACID**, garantindo a integridade dos dados e a confiabilidade das operações.

- Permite a manipulação avançada de dados. Esta ferramenta suporta uma ampla gama de funcionalidades, como índices, gatilhos e processos de armazenamento, que facilitam a manipulação eficiente dos dados armazenados.

- A possibilidade de manter a definição de micro serviços. Essa abordagem será explicada de forma mais detalhada em capítulos subsequentes, fornecendo uma compreensão aprofundada de como essa escolha se alinha com a arquitetura do sistema e oferece benefícios específicos para a implementação dos micro serviços.

Assim que estiver concluído, a empresa responsável planeia colocar o modelo de recomendações em produção. Uma vez que a empresa Bsolus já tem uma relação de confiança e experiência prévia a utilizar os serviços da *Amazon*, decidiu-se utilizar a ferramenta AWS *SageMaker Notebook Instance*, [Amazon, 2023], como infraestrutura para ter os serviços de recomendação a qualquer momento totalmente operacionais.

3.3 Armazenamento de recomendações

Tal como anteriormente referido, os resultados dos modelos de recomendação serão guardados numa coleção **PostgreSQL**. Um dos requisitos para este projeto é a possibilidade de executar os modelos simultânea ou individualmente, cabe ao cliente decidir, e, conforme o tipo de modelo, este guarda recomendações para um cliente ou para um produto em específico. Os resultados de cada modelo ficam

então previamente guardados em ficheiros separados, cada um com um nome específico e de fácil perceção. Exemplificando, os resultados do modelo de recomendação híbrido serão guardados no ficheiro *ma-data-engine-hybrid*.

Sempre que um modelo de recomendação é executado ao invés de apagar e reescrever todos os resultados de novo, o sistema verifica, a partir do *id* do artigo, quais foram modificados e/ou adicionados desde a última execução. Esta abordagem aumenta eficiência computacional, reduzindo o tempo e os recursos necessários para atualizar os resultados, uma vez que apenas reescreve os alterados, e minimiza a sobrecarga associada à reescrita de grandes conjuntos de dados, otimizando o desempenho do sistema.

A seguinte prática diz respeito à divisão de dados entre diferentes empresas que utilizam o mesmo modelo de recomendação. À alternativa de cada empresa ter o seu próprio ficheiro, todos os resultados são consolidados e armazenados num mesmo ficheiro na base de dados. Esta prática não só reduz espaço de armazenamento, como também simplifica a manutenção do sistema, já que a gestão centralizada de um único ficheiro facilita operações como *backup*, recuperação e manutenção.

A segurança é garantida através da introdução de uma variável denominada *tenant_id*. Esta variável protege os dados, assegurando que cada empresa só tenha acesso às informações associadas ao seu identificador único. Esta decisão não só preserva a confidencialidade dos dados, evitando o acesso não autorizado, mas também oferece um controlo de acesso granular, uma vez que cada empresa pode visualizar apenas os resultados associados à sua identificação única.

Cada documento é armazenado de forma independente para garantir a integridade dos dados e eliminar qualquer período de indisponibilidade nas recomendações. Isto significa que, mesmo durante a execução dos modelos de recomendação com base nos novos dados disponíveis, a plataforma permanece totalmente funcional e utilizável. Desta forma, é evitado que o sistema fique sem recomendações disponíveis.

Os modelos de recomendação seguem a seguinte estrutura:

```
        "id": Identificador único da recomendação no dataframe
        "product_id" || "client_id": Identificador hexadecimal único do
 produto/cliente associado à recomendação.

        "tenant_id": Identificador único, representa a empresa à qual
está associado a recomendação,

        "recommendations": Recomendações associadas ao artigo
específico identificado pelo "product\_id" ou "client\_id". É uma lista
```

```
de objetos que contém informações sobre cada recomendação. Cada
recomendação tem um grupo de informações especifico
```

Listing 3.1: Estrutura dos modelos de recomendação

3.4 Arquitetura do Sistema

A arquitetura de um projeto desempenha um papel crucial ao fornecer uma estrutura sólida que sustenta todos os aspetos do desenvolvimento. Em termos figurativos, a arquitetura é a espinha dorsal que define a organização e a interconexão dos seus componentes. À exceção de alguns componentes do projeto que podem evoluir com o tempo, a arquitetura é concebida no início e estabelece os alicerces imutáveis sobre os quais o restante do sistema é construído.

3.4.1 Síntese

1. Tendo a figura 6 como apoio visual, o processo começa com um pedido de inicialização por parte de um utilizador. Após o pedido, as informações providas pelo utilizador (modelo de recomendação que deseja iniciar), os dados providos pela empresa, nomeadamente os *dataframes* encomendas, clientes e produtos previamente guardados no PostgreSQL, são fornecidos ao sistema desenvolvido.

2. Assim que estão todos os dados necessários, o tratamento de dados é o próximo passo a decorrer. No conjunto de ficheiros serão incumbidos os dados fornecidos e a transformação dos mesmos, de modo a mitigar a fraca qualidade dos dados e reconhecer padrões futuramente valiosos.

3. O resultado do tratamento de dados será distribuído pelos ficheiros responsáveis pela **modelação** dos algoritmos **ML** previamente mencionados. Por motivos de lógica e transparência de código, cada modelo de recomendação é um ficheiro separado.

4. Como barreira de aprovação, os resultados dos modelos de recomendação são direcionados para o ficheiro de **avaliação** que, segundo as métricas previamente mencionadas, determinam se estes modelos atingem os objetivos inicialmente delineados.

5. Caso os resultados sejam satisfatórios, tanto os resultados das métricas como os resultados dos modelos serão guardados numa base de dados PostgreSQL fora do sistema de recomendações,

acessível ao utilizador que iniciou o processo. Caso os dados não cumprem os requisitos, reinicia-se o ciclo para o tratamento de dados e repete-se o processo.

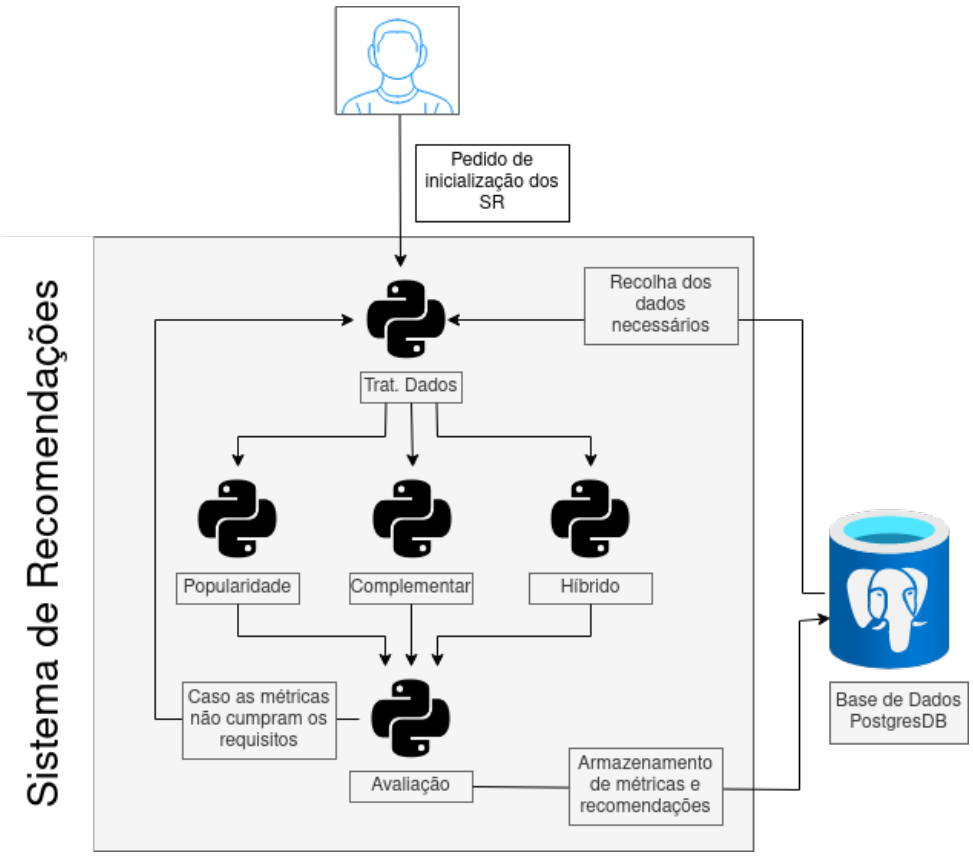


Figura 6: Representação da arquitetura do sistema de recomendações

3.4.2 Comunicação entre micro serviços

Apesar de não ser o foco da dissertação, é importante mencionar como o modelo de recomendações comunica com os restantes microsserviços, recebendo assim a informação necessária para a execução dos algoritmos de recomendação e posteriormente, enviar as recomendações criadas para a base de dados.

A comunicação entre microsserviços é feita a partir do *Redis*, sistema de armazenamento de dados em memória de alto desempenho que é amplamente utilizado para armazenar, gerir e recuperar informações rapidamente. A comunicação segue o padrão *Publish/Subscribe* (Pub/Sub) do **Redis**, modelo de troca de mensagens onde os remetentes (*publishers*) enviam mensagens para canais específicos, e os destinatários (*subscribers*) recebem as mensagens de canais aos quais estão inscritos.

3.4.3 Makefile

Makefile é um ficheiro de texto simples utilizado para construir e gerir projetos. Este é formado por uma ordem de comandos em que, ao ser executados, implementam, constroem e testam toda a ferramenta de recomendações. Posto isto, o Makefile deste projeto é constituído por:

- **make install:** Este comando é essencial e deve ser executado em primeiro lugar. Instala todas as bibliotecas necessárias para o funcionamento do programa. As bibliotecas estão listadas no arquivo "requirements.txt".
- **make format:** Sabendo que este seria um projeto de grandes dimensões, escolheu-se uma ferramenta de formatação de código que formata automaticamente o código *python* para seguir um estilo consistente, usando as regras estabelecidas no PEP8 (guia de estilo do *python*).
- **make lint:** A verificação de código, ou *linting*, é um processo para identificar erros e violações de estilo no código-fonte. Para este projeto utilizou-se a ferramenta *Pylint*, esta analisa o respetivo código e verifica questões como variáveis não definidas, importações não utilizadas e código duplicado.
- **make test:** Em *Python*, funções de teste são testes unitários onde o objetivo é testar unidades individuais de código (como funções ou classes) isoladamente do restante sistema. Posto isto, este comando promete aceder ao ficheiro de testes, que contem funções teste para as funções utilizadas ao longo da transformação de dados, e testar cada uma individualmente, provando que a função cumpre o desejado. É possível consultar um exemplo no apêndice função teste.

3.4.4 Arquivo de variáveis

O arquivo .env, habitualmente conhecido como arquivo de ambiente, é normalmente aplicado em serviços *Python* para armazenar variáveis de configuração sensíveis, específicas do ambiente ou que necessitem de constante modificação. Estas variáveis incluem informações como chaves de **API**, informações que não devem ser públicas ou configurações do sistema, entre as quais:

```
   # CONFIGURAÇÕES

N_CLUSTERS = Número de clusters a utilizar na segmentação de clientes (
   OPCIONAL)
CB_WEIGHT = Peso do modelo baseado em conteúdo em recomendações híbridas (
   OBRIGATÓRIO)
```

```
CF_WEIGHT = Peso do modelo de filtragem colaborativa em recomendações
  híbridas (OBRIGATÓRIO)
CL_WEIGHT = Peso do modelo de clustering em recomendações híbridas (
  OBRIGATÓRIO)

BOUGHT_PRODS = Incluir ou não produtos já comprados por um cliente nas
  suas recomendações (OBRIGATÓRIO)
LAST_MONTHS_ORDERS = Número de meses a serem considerados no histórico de
  pedidos ao produzir recomendações (OPCIONAL)
EVALUATE = Booleano para indicar se as métricas devem ser calculadas para
  avaliar a precisão dos modelos, sempre que são executados (OPCIONAL)

CPUS = Número de CPUs utilizadas (OPCIONAL)
TOPN = Recomendar apenas os primeiros n produtos no modelo de popularidade
   (OPCIONAL)

# COMUNICAÇÃO
POSTGRES_DB = base de dados para conectar ao POSTGRES
POSTGRES_USER = utilizador para conectar ao POSTGRES
POSTGRES_PASS = palavra-passe para conectar ao POSTGRES
POSTGRES_PORT = porta para conectar ao POSTGRES

REDIS_PORT = porta do redis para conectar ao MA
REDIS_URI = URI do redis para conectar ao MA
REDIS_HOST = anfitrião do redis para conectar ao MA
REDIS_CHANNEL = canal do redis para conectar ao MA
```

Listing 3.2: Representação do .env

Este arquivo está organizado em duas secções: comunicação e configurações. A secção comunicação centra-se nas variáveis responsáveis pela comunicação do sistema de recomendações com a restante ferramenta, ou seja, com os restantes micro serviços (*redis*) e a base de dados onde se irá guardar os resultados dos modelos (PostgreSQL)

Quanto à parte configuração, reúne-se as variáveis que podem e devem ser facilmente modificadas, por um responsável, conforme as necessidades do projeto. Determinadas variantes estão marcadas como opcional, por possuírem um valor padrão implementado. Por exemplo, caso não seja atribuído um valor à variável "*n_clusters*", o modelo *Clustering* está programado para atribuir um valor a partir do *Elbow Method*, posteriormente mencionado e explicado. Caso a inconstante "*last_month_orders*" não possua

um valor, serão considerados todos os dados recolhidos da base de dados.

No contexto da dissertação, a utilização de um arquivo .env é benéfico por que separa configurações sensíveis do código-fonte e dá o fácil acesso a variáveis que podem ser modificadas para aumentar a eficácia dos modelos, que de outra forma estariam implementadas *hard coded*. Para manter a confidencialidade, o arquivo .env foi incluído no arquivo .gitignore para garantir que as informações confidenciais não sejam compartilhadas.

3.5 Documentação

A documentação é fundamental num projeto de programação, por ajudar a garantir a clareza e compreensão do código. Esta fornece explicações, guias e exemplos que facilitam o entendimento das funcionalidades implementadas. Ao descrever as *interfaces*, funções e classes, a documentação permite que os programadores entendam o propósito e o comportamento de cada parte do código.

Para qualquer questão, todo o projeto desenvolvido está devidamente documentado na ferramenta *Confluence*,[Gonçalo Almeida, 2022], uma parte essencial da plataforma de gestão de desenvolvimento *Atlassian*, utilizada pela Bsolus. Nesta documentação, encontra-se uma descrição detalhada da proposta de desenvolvimento, do contrato de dados estabelecido, de todas as regras que devem ser seguidas durante a implementação e, é claro, de todas as especificações dos desenvolvimentos realizados. Isto facilita significativamente a integração de novos colaboradores, bem como o trabalho dos administradores do produto, que podem consultar todo o histórico de estratégias e decisões que resultaram no sistema de recomendações atualmente disponível.

Para uma documentação mais detalhada, função por função, utilizou-se a ferramenta *SPHINX* para transportar a definição de cada função para uma página *Confluence* devidamente identificada, [Almeida, 2022].

Capítulo 4

Implementação

A implementação de qualquer sistema complexo obriga a uma sequência estruturada de etapas que um indivíduo deve seguir para tomar a melhor decisão possível entre as várias alternativas disponíveis. O propósito principal de um processo *Data Mining* é ajudar a tomar decisões informadas e eficazes, especialmente em situações complexas ou de grande relevância. Como previamente apontado, o processo escolhido foi **CRISP-DM** e seguir esta estrutura ajudará a identificar e, posteriormente, a resolver problemas que este processo possa trazer.

4.1 Compreensão de dados

Nesta fase inicial pretende-se recolher e compreender, através do método **EDA** previamente explicado, os dados fornecidos. A identificação de possíveis relações, padrões e anomalias entre a informação recolhida terá um impacto positivo nos modelos de recomendação. O propósito é compreender integralmente os dados, descobrir eventuais relações e tendências e resolver qualquer problema na qualidade dos dados antes de avançar para as fases subsequentes.

4.1.1 Exploração de Dados: Análise Preliminar

A otimização dos três *dataframes* inicia-se com a identificação de qualquer tipo de inconsistências que possam incorretamente influenciar os modelos de recomendação. Assim sendo, é imperativo que qualquer valor incongruente seja devidamente tratado para garantir a integridade dos modelos.

Falta de dados

A falta de dados em vários atributos durante o processo de recolha de informações sobre clientes, produtos e/ou encomendas representa um dos problemas mais frequentes. Esta carência pode surgir tanto devido à negligência dos utilizadores como à omissão deliberada por parte das empresas envolvidas. Dependendo

da situação, para a entrega do *dataframe* final, a resolução deste problema envolverá uma das seguintes opções: remoção da linha, remoção da coluna ou preenchimento do valor em falta.

Outliers

Não menos relevante é a presença de *outliers*, que representam valores atípicos tendo em conta os restantes valores da coluna. No entanto, a existência de *outliers* é um caso delicado. Por um lado, a presença de *outliers* nos dados fornecidos pode gerar recomendações menos precisas, valores divergentes podem distorcer o modelo. Por outro lado, os mesmos valores podem proporcionar informação valiosa e contribuir para a eficácia do modelo, tudo depende da coluna em questão.

A decisão sobre como tratar os *outliers* de uma específica coluna irá depender do contexto. Por exemplo, dado que, no contexto apresentado, os produtos disponíveis incluem cápsulas de café (um produto relativamente barato) e máquinas de café (um produto relativamente caro) é possível que esses valores se manifestem como *outliers* quando, na realidade, não o são.

Um valor será identificado como *outlier* caso o seu *z_score_cutoff* seja maior ou igual que 3. O *Z-Score* é uma medida estatística que indica a quantos desvios padrão um determinado valor está da média. O *Z-Score* é calculado individualmente para cada coluna a partir da seguinte fórmula:

$$Z = \frac{(x - \alpha)}{\sigma}$$

Formula para calcular outliers

- Z : *Z-score*;
- x : Valor da coluna;
- α : Média dos valores da coluna;
- σ : Desvio padrão dos valores da coluna.

Assim, valores que estejam além de três desvios padrão da média são considerados *outliers* e serão removidos do *dataframe* durante o processo de análise exploratória de dados para garantir a integridade das análises e modelos desenvolvidos.

Imparcialidade dos dados

A possibilidade de haver uma distribuição desbalanceada de informações de um determinado campo, seja este geográfico, género, idade, entre outros, pode levar a influenciar os resultados negativamente nos modelos de recomendação. Tal falta de imparcialidade pode originar uma experiência de recomendação

restrita e prejudicar a diversidade de opções apresentadas aos utilizadores. Ademais, pode reforçar estereótipos existentes ou acentuar desigualdades sociais, ao não considerar equitativamente as preferências e necessidades de grupos sub-representados.

4.1.2 Contrato de dados: atributos necessários

Tal como na esmagadora maioria das empresas destinadas a *e-commerce*, a estrutura das empresas alvo consiste em duas entidades distintas: **produtos e clientes**. Estas entidades, ao interagirem entre si, criam uma terceira entidade denominada **encomendas**, que regista as transações efetuadas pelos clientes relativamente aos produtos.

Um dos principais objetivos estabelecidos neste projeto foi garantir que os modelos de recomendação fossem altamente adaptáveis para diferentes plataformas de negócio, possibilitando que diferentes empresas utilizassem a mesma transformação de dados e os respetivos modelos sem qualquer interferência por parte dos responsáveis. A universalidade e reutilização dos modelos simplifica a implementação e reduz a dependência de conhecimentos técnicos para a utilização da ferramenta de recomendação. Para concretizar esta visão, foi desenvolvido um contrato de dados que todas as empresas devem seguir para conseguirem utilizar o nosso sistema de recomendação. Este contrato estabelece um conjunto de campos de dados obrigatórios que devem ser fornecidos pelas empresas na introdução dos seus valores. Ao seguir estas diretrizes, garante-se a uniformidade e consistência dos dados, permitindo que o modelo de recomendação seja aplicado sem dificuldades em qualquer cenário. A criação do contrato de dados possibilitou uma padronização na forma como os dados são recolhidos, estruturados e disponibilizados para o sistema. Isto viabiliza a generalização dos algoritmos e a abstração de peculiaridades de cada negócio, tornando a implementação do modelo de recomendação mais ágil e menos suscetível a erros humanos.

Para a realização deste projeto, a empresa Bsolus, fornecerá dados reais sobre os produtos, clientes e encomendas da empresa DeltaQ Portugal, líder nacional no setor de café, cápsulas e máquinas de café. Para o desenvolvimento prático deste projeto, os dados fornecidos terão cerca de 400 produtos, 250 mil clientes e quase 200 mil encomendas, esta informação será detalhada no capítulo seguinte. Os contratos estipulados para a entidade Produtos (2), a entidade Clientes (3) e a entidade Encomendas (4) pretendem abranger e informar todos e qualquer atributo que possam, de alguma forma, influenciar os modelos de recomendação.

Entidade	Atributos	Tipo	Descrição
	id	string	Identificador da linha do dataframe
	product_id	string	Identificador hexadecimal do produto
	tenant_id	string	Identificador da empresa
Produtos	name	string	Nome do produto
	stock	int	Quantidade de produtos disponíveis para venda
	pricing	list [int]	Conjunto de informações sobre o preço do produto
	status	string	Estado do produto na plataforma
	created_at	date	Data da criação do produto
	updated_at	date	Data da última modificação do produto
	region	string	Região do produto
	extra_fields	list [string]	Informações extra que podem ser adicionadas
	sku	int	Número atribuído acompanhar os níveis de stock
	ean	string	código de barras padronizado

Tabela 2: Contrato de dados implementado dos produtos

Entidade	Atributos	Tipo	Descrição
	id	string	Identificador da linha do dataframe
	user_id	string	Identificador hexadecimal do produto
	tenant_id	string	Identificador da empresa
	name	string	Nome do produto
Clientes	email	string	E-mail do cliente
	phone_number	int	Número do cliente
	status	string	Estado do cliente na plataforma
	gender	string	Género do cliente
	birth_date	date	Data de nascimento do cliente
	created_at	date	Data de criação do cliente
	updated_at	date	Data de modificação do cliente
	status	string	Estado do cliente
	subscription	string	Subscrição do cliente
	address	list [string]	Morada completa do cliente
	market	string	Mercado do cliente
	extra_fields	list [string]	Informações extra sobre o cliente

Tabela 3: Contrato de dados implementado dos clientes

Entidade	Atributos	Tipo	Descrição
	id	string	Identificador da linha do dataframe
	order_id	string	Identificador hexadecimal da encomenda
	order_number	string	Número da encomenda
	user_id	string	Identificador hexadecimal do cliente
Encomendas	tenant_id	string	Identificador hexadecimal da empresa
	total	int	Valor total da encomenda
	discount	string	Desconto relativo à encomenda
	coupons	string	Cupões utilizados na encomenda
	currency	string	Moeda utilizada
	created_at	date	Data de criação da encomenda
	updated_at	date	Data de modificação da encomenda
	status	string	Estado da encomenda
	shipping	list [string]	Endereço para onde a encomenda foi enviada
	payment	list [string]	Tipo de pagamento utilizado
	ship_info	string	Informação sobre o Endereço
	products	list [string]	Informação sobre o(s) produto(s) comprados
	extra_fields	list [string]	Informação extra sobre a encomenda

Tabela 4: Contrato de dados implementado das encomendas

4.2 Preparação dos dados

4.2.1 Redução de dados

A etapa inicial desta terceira fase envolve a eliminação de dados não essenciais. Esses dados incluem *outliers*, valores em falta e colunas irrelevantes para os modelos de recomendação. Tendo em conta a percentagem de valores em falta/*outliers* numa coluna, segue-se a seguinte norma:

- **Até 20%: descartar exceções:** Todas as entradas no *dataframe* que contenham este tipo de dados são excluídas. Esta taxa de corte é considerada aceitável e não compromete a integridade do *dataframe*, especialmente quando lidamos com um conjunto de dados de grande dimensão. Ao remover as entradas com *outliers* ou valores ausentes em colunas específicas, garante a qualidade

dos restantes dados. Este limite foi estabelecido pela equipa responsável após criteriosa avaliação.

- **Entre 20% até 50%: preencher com o valor médio:** Caso a percentagem de valores em falta esteja compreendida entre estes valores e a coluna seja marcada como interesse, os valores em falta serão preenchidos com o valor médio (ou o mais comum). A ideia é influenciar o mínimo possível na decisão final, colunas como género ou idade, questões que muita gente decide omitir, são geralmente preenchidos com a média. No caso de *outliers* é preenchido com o máximo (ou mínimo aceitável).

- **Mais de 50%: eliminar coluna:** Na situação em que a percentagem de valores em falta ultrapasse metade dos valores, a coluna perde o seu uso e removida e não influenciará os modelos recomendações.

Quanto às colunas, foram eliminadas aquelas cujos valores não podem ser equiparados entre si, que não trouxessem vantagens aos modelos de recomendação ou que tivessem um número elevado de valores em falta. Por exemplo, o número de telemóvel do cliente, embora necessário para fins de publicidade, não apresenta nenhum benefício relevante aos modelos, a coluna *sku* (*Stock keeping unit*) apresenta demasiados valores em falta e tabela *ean* (*European article number*) apresenta valores que não são equiparáveis entre si.

Ao proceder com a remoção de dados não essenciais, pretendeu-se assegurar a qualidade e efetividade do processo de preparação dos dados, visando obter resultados mais robustos e confiáveis nas etapas subsequentes da análise de recomendação.

4.2.2 Pré-processamento de Dados

Após remover a informação irrelevante e/ou prejudicial, o próximo passo consiste em extrair o máximo de conhecimento possível das colunas restantes. Esta etapa envolve a aplicação de técnicas de engenharia de características e análise estatística visando desvendar informações valiosas e padrões relevantes, essenciais para a construção de modelos de recomendação eficazes.

Geração de novas colunas

O próximo passo traduz-se em dividir colunas que contenham múltiplos tipos de informação (identificadas nas tabelas como *list* [*string*] em colunas individuais, em que cada uma representa um valor específico. A figura 9 representa a fragmentação da coluna "*address*", esta será subdividida em 4 colunas: "*address.zip*", "*address.country*", "*address.city*", "*address.locality*".

No caso da data de nascimento dos clientes, embora seja uma informação interessante para fins publicitários, esta não será tão útil para um algoritmo de recomendação. No entanto, a partir da data, podemos determinar a idade, ou ainda melhor, em faixas etárias:

- **Criança:** Até aos 18 anos
- **Jovem:** Dos 18 aos 29 anos
- **Adulto jovem:** Dos 30 aos 39 anos
- **Adulto:** Dos 40 aos 49 anos
- **Adulto sénior:** Dos 50 aos 59 anos
- **Sénior:** A partir dos 60 anos

Com esta nova informação, podemos fornecer ao algoritmo uma nova forma de relacionar os clientes entre si. Exceções à parte, clientes com uma idade registada de menos de 18 anos não serão tidos em conta. Enquanto a figura 7 representa as idades dos clientes sem qualquer tipo de tratamento, originando várias colunas e dificultando o relacionamento entre os clientes. Já a figura 8 demonstra a relação entre os clientes a partir da sua faixa etária.

Outra funcionalidade desenvolvida a partir do *dataframe* encomendas foi a criação de uma nova coluna "*season*". Esta nova funcionalidade determina a estação do ano em que foi realizada a encomenda, ou seja, cada entrada do *dataframe* terá uma das quatro opções: verão, inverno, outono ou primavera. Esta referência será posteriormente utilizada para construir um perfil específico tanto para o cliente quanto para o produto.

Um dos passos com maior relevância compôs-se na **fragmentação das encomendas** que continham dois ou mais produtos em encomendas individuais. As encomendas que agrupam múltiplos produtos são divididas em entradas separadas no *dataframe* encomendas. Ou seja, caso um cliente tenha realizado uma encomenda com *n* produtos, a mesma foi subdividida em *n* entradas no *dataframe* encomendas, em que todos os dados serão idênticos, exceto a coluna produtos, que em vez de ter uma lista de produtos, terá apenas o produto n.

Esta abordagem permite a crítica e o processamento individual de cada produto, o que contribuirá para a eficácia do algoritmo **ML** em questão. Esta estratégia é de extrema importância para recolher as características distintas de cada produto, sem que outro produto na mesma encomenda influencie o resultado, garantindo, assim, maior personalização e direcionamento nas recomendações oferecidas aos

clientes. Ademais, a subdivisão das encomendas facilita a integração dos modelos de recomendação com outros conjuntos de dados ou características específicas de cada produto.

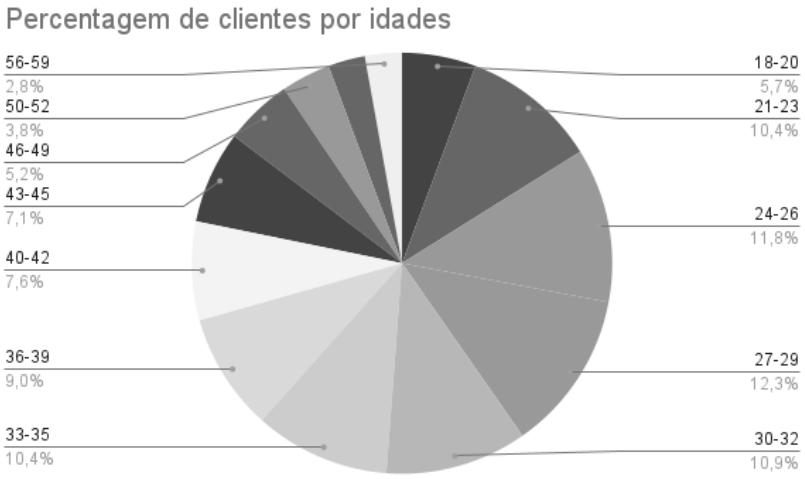


Figura 7: Representação das idades dos clientes

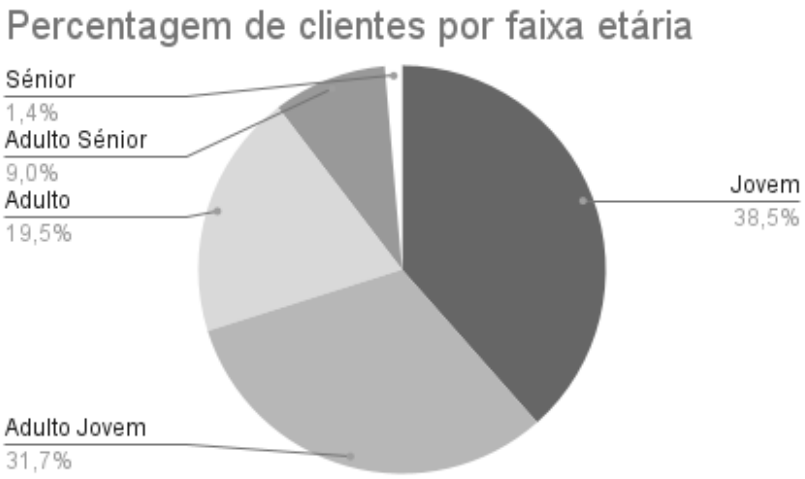


Figura 8: Representação das idades dos clientes por faixa etária

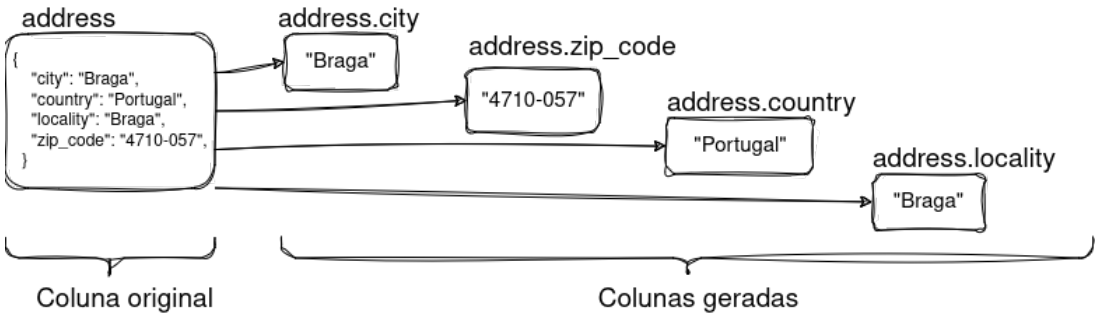


Figura 9: Fragmentação da coluna address

Normalização e categorização de colunas

Um processo *standard* em qualquer projeto que envolva uma quantidade significativa de dados é a normalização de colunas numéricas (identificadas como *int*) e categorização de colunas constituídas por um conjunto limitado de texto (identificadas como *string*). Estas medidas são vulgarmente adotadas para alcançar maior uniformidade e coerência na representação dos dados, tornando-os comparáveis e interpretáveis de forma mais eficaz entre si. A normalização de uma coluna numérica, ou seja, a transformação dos seus valores para uma mesma ordem de magnitude, ajusta os valores a uma escala comum, eliminando possíveis intervalos entre valores inerentes às diferentes grandezas presentes nos dados.

A categorização das colunas que apresentem um conjunto limitado de *strings* possibilita agrupar e classificar os dados em categorias distintas. Esse procedimento é especialmente relevante quando as informações se referem a atributos discretos ou qualitativos, como tipos de produtos, géneros, endereços ou até faixa etária. Ao categorizar estes valores, simplifica a identificação de padrões, relações e tendências.

Atribuição de um score

Após a transformação dos *dataframes*, é necessário arranjar uma métrica capaz de avaliar os produtos imparcialmente a partir das experiências que os clientes tiveram com os mesmos. A conduta escolhida para mitigar este problema foi o cálculo de um *score* a cada encomenda individual, em que, posteriormente, o *score* de um produto será média do *score* das suas encomendas.

A questão que se sobrepõe é saber quais as variáveis a considerar para avaliar os produtos de forma justa e imparcial. Tendo em conta os dados disponíveis em cada encomenda, o *score* será calculado a partir das seguintes variáveis:

- **product.quantity**: Número de vezes que o cliente encomendou o respetivo produto na mesma encomenda;

- **product.hits_count**: Número de vezes em que o determinado produto surgiu numa encomenda, independentemente do cliente;

- **status.strength**: O estado no qual a encomenda se encontra. Por exemplo, encomendas que são canceladas indicam um nível de satisfação do cliente mais baixo, enquanto as encomendas confirmadas representam o oposto, ou seja, um cliente mais satisfeito.

Sendo que cada encomenda tem um *score* associado, não só permite avaliar os produtos como também captar a experiência individual de cada cliente. Dado que os valores nas colunas "*quantity*" e "*hits_count*" podem ser bastante elevados, decidiu-se atribuir a cada uma delas um peso de 20% enquanto a coluna "*status*" (normalizada entre 0 e 1) recebeu os restantes 60% como peso:

$$\begin{aligned} \textbf{Score} = & df["product.hits_count"] \times 0.2 + \\ & df["product.quantity"] \times 0.2 + \\ & df["product.status.strength"] \times 0.6 \end{aligned} \tag{4.1}$$

Agora que cada encomenda individual tem um *score* atribuído, calcula-se a média do *score* para cada produto e posteriormente adicionado ao *dataframe* dos produtos, fornecendo assim uma pontuação a cada produto.

No entanto, surge um obstáculo: como lidar com produtos que nunca foram comprados ou que não possuem registos de encomendas. Uma vez que a fórmula utiliza variáveis provenientes da interação entre produto-cliente, produtos que não tenham sido encomendados são descartados nesta variável *score*. Para enfrentar esta situação, é necessário calcular um *score* diferente para estes produtos. Para o cálculo deste *score* secundário consideraram-se intrínsecas do produto, como preço, descrição e quantidade em *stock*. Assim, mesmo produtos sem histórico de vendas podem ser avaliados com base nas informações disponíveis, proporcionando uma visão completa do catálogo de produtos.

Após a transformação dos dados provenientes do contrato de dados previamente mencionado, as tabelas 5, 6, 7 representam a informação que segue para o passo seguinte.

Entidade	Atributos	Tipo	Descrição
	product_id	string	Identificador hexadecimal do produto
	tenant_id	string	Identificador da empresa
Produtos	name	string	Nome do porduto
	stock	int	Quantidade disponivel para venda
	created_at	date	Data de criação do produto
	updated_at	date	Data da última modificação do produto
	pricing.sales_price	float	Preço atual do produto
	attributes	string	Características do produto
	category	string	Categoria do produto
	ACTIVE	int	Estado do produto
	Score	float	Valor do produto

Tabela 5: *Dataframe* Produtos atualizado

Entidade	Atributos	Tipo	Descrição
	client_id	string	Identificador hexadecimal do cliente
	tenant_id	string	Identificador da empresa
Clients	name	string	Nome do produto
	birth.date	date	Data de nascimento fornecida pelo cliente
	birth.age	int	Idade do cliente segundo a sua data de nascimento
	birth.age.range	int	Faixa etária do cliente segundo a sua data de nascimento
	gender	int	Género identificado pelo cliente
	created_at	date	Data de criação da conta do cliente
	updated_at	date	Data da última modificação na conta do cliente
	email	string	Email do cliente
	address.city	string	Cidade onde reside o cliente
	address.country	string	País onde reside o cliente
	address.locality	int	Localidade onde reside o cliente
	address.zip_code	float	Código postal onde reside o cliente
	ACTIVE	int	Estado do cliente

Tabela 6: *Dataframe* Clientes atualizado

Entidade	Atributos	Tipo	Descrição
	order_id	string	Identificador hexadecimal da encomenda
	tenant_id	string	Identificador da empresa
Encomendas	client_id	string	Identificador hexadecimal do cliente
	product_id	string	Identificador hexadecimal do produto
	created_at	date	Data de criação da encomenda
	updated_at	date	Data da última modificação da encomenda
	email	string	Email do cliente
	Total	float	Custo total desta encomenda
	product.quantity	int	Quantidade do mesmo produto encomendada
	product.subtotal	float	Custo total do produto em questão
	product.categories	float	Categoria do produto em questão
	season	int	Estação na qual o produto foi adquirido
	status.strength	int	Estado da encomenda
	score	float	Valor da encomenda

Tabela 7: *Dataframe* Encomendas atualizado

4.2.3 Aumento de performance

Após a primeira execução total do tratamento de dados, observou-se que o desempenho do algoritmo estava muito abaixo do esperado, ou seja, o tempo de execução era bastante extenso para ser um projeto viável e eficiente. A razão por detrás desta fraca execução era devido à iteração sequencial dos *dataframes*, isto é, previamente comprovado que os *dataframes* são de grandes dimensões, o algoritmo percorria entrada por entrada, sequencialmente, o que era extremamente custoso e demorado. Como solução do problema, consideraram-se três opções:

- **Amostra de dados:** Durante a fase de testes e acertos utilizar apenas uma amostra selecionada de dados. Isto diminuiria drasticamente a dimensão dos *dataframes* e, consequente, o tempo de execução reduzia para um tempo aceitável. No entanto, esta opção não resolveria o problema, pois o modelo final permite às plataformas correrem os modelos de recomendação com os dados que desejarem, caso estes sejam de grande dimensão, voltaríamos ao problema inicial.

- **Armazenamento em cache:** Após a execução do tratamento de dados, armazenar em me-

mória transitória os resultados, ou seja, os *dataframes* finais e aceder sempre a esta informação pré-processada sempre que se desejar correr os modelos de recomendação. Apesar de parecer aliciante obter os dados necessários para os modelos imediatamente, é necessário ter em conta que as plataformas podem adicionar nova informação (seja produtos, clientes ou encomendas) a qualquer momento e esta poderá ter influência nos modelos de recomendação, tornando a informação previamente armazenada, totalmente obsoleta.

- **Processamento paralelo:** Sempre que se executar o processo de tratamento de dados, utilizar técnicas de processamento paralelo capaz de realizar operações em vários núcleos de CPU, ao mesmo tempo. Esta opção soluciona os desafios previamente apresentados, pois sempre que a plataforma quiser correr os algoritmos de recomendação este inicia o processo de raiz, com os dados atualizados, sejam estes um conjunto específico ou a adição de nova informação.

Face às hipóteses, é evidente que o melhor caminho a seguir é o **processamento paralelo**, nomeadamente **threads**. *Threads* são unidades básicas de execução num programa, que permitem executar múltiplas tarefas simultaneamente. Ao dividir o trabalho em *threads* independentes, cada uma pode executar em paralelo, permitindo que várias operações sejam realizadas ao mesmo tempo. Ao utilizar *threads* na iteração dos *dataframes*, em vez de processar cada entrada sequencialmente sempre no mesmo *thread*, o *dataframe* é fragmentado em secções com *N* entradas e cada uma atribuída a uma *thread* livre, que posteriormente executam todos ao mesmo tempo, em paralelo. Tal como esperado, esta abordagem diminuiu o tempo de execução significativamente, o que contribuiu para a eficácia do modelo e assim melhorar a experiência do utilizador.

4.3 Modelação

Até este momento foi explorado o processo de preparação de dados, no seguinte subcapítulo utilizaremos os resultados para criar e aperfeiçoar os nossos modelos de recomendação. É aqui que damos o salto qualitativo, aproveitando ao máximo os dados preparados para melhor servir as necessidades dos nossos futuros clientes.

Neste capítulo, discutiremos a implementação de um modelo híbrido que combina Filtragem Colaborativa, Filtragem Baseada no Conteúdo e uma técnica de *clustering*. Adicionalmente, adicionamos dois modelos à nossa abordagem: Popularidade e Complementar de produtos, enriquecendo ainda mais a nossa capacidade de fornecer recomendações de alta qualidade aos nossos utilizadores.

4.3.1 Popularidade

Um algoritmo de recomendação baseado na popularidade é uma abordagem simples que se baseia na popularidade dos produtos para gerar recomendações. Ao contrário dos restantes modelos, este não considera as preferências individuais dos clientes, mas sim avalia a experiência de todos como um só e fornece uma lista igual para todos os clientes, independentemente de quem são ou de como interagiram com a plataforma em questão. A lógica de negócio afirma que artigos populares são populares por alguma razão, se a maioria dos clientes gosta de um produto é provável que novos clientes irão gostar também.

Este simples algoritmo, para determinar a popularidade de cada produto, tem apenas em conta a variável previamente apresentada no subcapítulo 4.2.2 , *score*. Esta variável tem em conta os atributos que a equipa considera que faz um produto popular. Como resultado, é fornecida uma lista, por ordem decrescente, com os produtos em função do seu *score*.

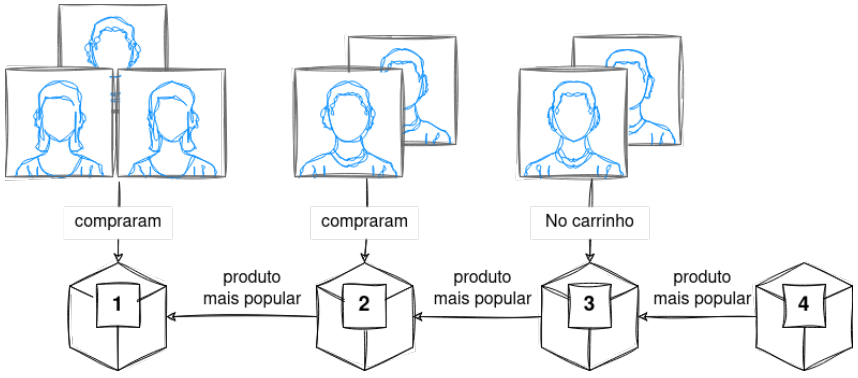


Figura 10: Exemplo do modelo popularidade

Na representação visual, figura 10, do modelo de popularidade, é possível compreender que a listagem de produtos mais populares segue a ordem: $1 \rightarrow 2 \rightarrow 3 \rightarrow 4$. O produto 4, uma vez que ainda não possui um histórico de compras, é o produto menos popular, pois só terá em conta o *score* secundário. O *score* entre o produto 2 e 3 difere porque, no produto 3, os clientes ainda não finalizaram a compra, diminuindo o campo *status.strength*. Por fim, entre o produto 1 e 2, difere a quantidade de pessoas que compraram o produto, uma vez que o valor de *product.hits_count* é maior no produto 1, este é classificado como o produto mais popular da lista.

4.3.2 Modelo Complementar

Ao contrário de qualquer modelo previamente discutido, a filtragem por produtos complementares é um tipo de sistema de recomendação que sugere produtos que complementam ou combinam bem com um

produto específico. Em vez de se concentrar apenas na semelhança, a filtragem de produtos complementares analisa as relações que diferentes artigos possam ter e recomenda os que aprimoram o uso ou a experiência de produto principal. De uma forma sucinta, o algoritmo cria associações entre produtos que um determinado cliente selecionou e quais os produtos que costumam ser comprados simultaneamente. Para alcançar este tipo de recomendação, o modelo utiliza regras de associação entre pedidos para criar conexões entre produtos. Um caso comum, será recomendar uns auscultadores e/ou uma capa de telemóvel a um cliente que tenha um telemóvel no seu carrinho de compras.

Rule	Support	Confidence	Lift
$A \Rightarrow D$	2/5	2/3	10/9
$C \Rightarrow A$	2/5	2/4	5/6
$A \Rightarrow C$	2/5	2/3	5/6
$B \& C \Rightarrow D$	1/5	1/3	5/9

Figura 11: Exemplo de filtragem Complementar

Observando a figura 11, na primeira regra: A → D podemos observar que o suporte é de 2/5 (0,4). Isto significa que, em 5 pedidos, sempre que A (antecedente) é comprado, D (consequente) também é comprado. Quanto à coluna confiança, em 3 pedidos em que A aparece, em 2 deles, D foi comprado. Para cada produto, selecionamos os pedidos que contêm esse produto e desenvolvemos regras de associação, utilizando o algoritmo *Apriori*. Este algoritmo retorna um conjunto de regras e o seu suporte. Filtramos as regras acima de um limiar configurável e recomendamos o consequente em cada regra.

4.3.3 Clustering

Um modelo *clustering* é uma técnica de análise de dados que, nesta dissertação em específico, agrupa os clientes conforme as suas características. O objetivo será determinar padrões naturais nos clientes que podem não ser imediatamente visíveis. No entanto, esta abordagem levanta uma nova questão: quais são os atributos dos clientes em que nos devemos focar para encontrar semelhanças entre os mesmos. Para mitigar este problema imparcialmente e não subjetiva, criou-se uma tabela de correlação entre as colunas presentes no *dataframe* finalizado dos clientes e obtiveram-se as seguintes colunas como as mais

eficazes:

- Faixa etária;
- Idade;
- Localidade;
- Cidade;
- Género.

No mesmo contexto, foi optado por implementar a estratégia *K-means*. Esta técnica é amplamente reconhecida e aplicada na análise de dados para agrupar elementos com base na sua proximidade e semelhança. Entre muitas, *K-means* foi a abordagem que se destacou por ser algoritmo eficiente que consegue lidar com grandes volumes de dados de forma relativamente rápida e pela sua facilidade de implementação.

Contudo, de acordo também com [Costa, 2021], vale mencionar que o sucesso da aplicação do *K-means* depende do número selecionado de *clusters*(K), ou seja, quantidade de grupos em que o cliente pode se relacionar, relembro que cada cliente apenas entra em um e só um grupo. Posto isto, utilizou-se o **Elbow method** para determinar o valor mais apropriado de K. O método do cotovelo, figura 12, envolve a execução do algoritmo *K-means* para diferentes valores de K e, em seguida, a avaliação da variação da dispersão dos dados relativamente ao número de *clusters*. O ponto em que a redução dessa variação começa a diminuir acentuadamente é o chamado "cotovelo" e sugere o número ótimo de *clusters*. Com os clusters delineados e completos, isto é, cada cliente pertencer ao grupo que apresente mais semelhanças a si, é corrido o modelo de popularidade anteriormente mencionado em cada um dos *clusters* apresentados. K listas dos produtos mais populares.

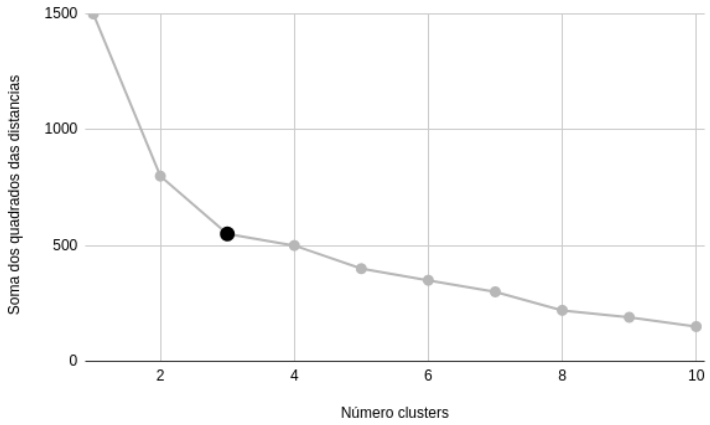


Figura 12: Número ideal de clusters

4.3.4 Filtragem Baseada em Conteúdo

Tal como previamente explicado, a filtragem baseada em conteúdo é uma abordagem que recomenda produtos com base nas características dos produtos previamente escolhidos pelo utilizador. Para a construção deste algoritmo utilizou-se a técnica **TF-IDF**, [Van Meteren and Van Someren, 2000].

O **TF-IDF** permite avaliar a importância de uma palavra num documento, num conjunto de documentos. Esta técnica combina duas medidas fundamentais: a frequência da palavra (TF) e o inverso da frequência do documento (IDF):

1. **Frequência da Palavra, TF**: A frequência da palavra desempenha um papel crucial na avaliação da relevância de um produto numa plataforma *e-commerce*. Esta é calculada ao dividir o número de vezes que um produto específico é mencionado relativamente ao número total de produtos no catálogo.

$$TF_t, d = \frac{Numero\ de\ vezes\ que\ a\ palavra\ x\ aparece\ no\ documento\ d}{Numero\ total\ de\ palavras\ no\ documento\ d}$$

2. **Inverso da Frequência do Documento, IDF**: O IDF avalia a raridade de um produto em todo o catálogo. Quanto mais raro o produto, maior será o valor IDF lhe atribuído. O cálculo do IDF considera quantos produtos mencionam ou estão relacionados a um artigo específico.

$$IDF_t = log \frac{Nmero\ total\ de\ documentos\ no\ corpus}{Nmero\ de\ documentos\ que\ contm\ a\ palavra\ t}$$

3. **TF-IDF**: Por fim, multiplica-se o TF pelo IDF. Isto resulta numa pontuação que reflete tanto a frequência da palavra no documento quanto a sua raridade no documento. Quanto maior o valor de **TF-IDF** para uma palavra num documento, mais importante essa palavra é considerada para esse documento em particular.

$$T_f - IDF_{t,d} = TF_{t,d} * IDF_t$$

Neste contexto, o **TF-IDF** será aplicado para avaliar a relevância de palavras-chave nas descrições de produtos, nas avaliações dos clientes e em possíveis outros textos relacionados com os produtos. Esta técnica permite ao sistema de filtragem baseada em conteúdo identificar produtos relevantes com base nas preferências do utilizador, melhorando a precisão das recomendações e, por conseguinte, a satisfação do cliente. Na tabela seguinte podemos ver as palavras com mais peso para um cliente aleatório.

	Token	Relevância
0	cápsulas	0.331234
1	café	0.220987
2	vermelho	0.2200456
3	médio	0.19485
4	máquina	0.12345
5	azul	0.10985

4.3.5 Filtragem Colaborativa

A Decomposição em Valores Singulares, ou **SVD**, [Pazzani and Billsus, 2007], é uma técnica matemática fundamental utilizada para analisar e representar matrizes de dados de uma maneira mais eficaz. A ideia central desta abordagem consiste em decompor uma matriz em três partes principais, resultando numa representação mais compacta dos dados originais. Esta é composta pela seguinte fórmula e consequente explicação:

$$A = USV^T$$

1. **Matriz Original (A):** Começamos com a matriz de avaliações de utilizadores e produtos.De dimensões *m x n* esta matriz, em geral, possui muitos espaços em branco, pois nem todos os utilizadores avaliam todos os produtos. Em cada célula, temos uma avaliação de um utilizador para um produto, ou pode ser um valor nulo quando não há avaliação disponível.

2. **Matriz de Valores Singulares (S):** A segunda parte é uma matriz diagonal, de dimensões *r x r* chamada matriz de valores singulares. Ela contém os valores singulares, os quais são números não negativos dispostos em ordem decrescente na diagonal principal. Esses valores singulares representam a importância relativa das características subjacentes aos dados.

3. **Matriz de Vetores Singulares (U e V):** A terceira parte consiste em duas matrizes de vetores singulares. A matriz U, de dimensões *m x r*, contém os vetores de singular para os utilizadores, enquanto a matriz V, de dimensões *n x r* contém os vetores de singular para os produtos. Esses vetores representam as relações entre os utilizadores e os produtos, com base nos valores singulares.

O principal conceito por trás do **SVD** é que, ao representar a matriz de avaliações de forma mais compacta usando os valores singulares e os vetores singulares, é possível fazer uma aproximação da matriz original. Este método torna-se especialmente útil em sistemas de recomendação, onde é possível prever as avaliações que um utilizador dê a produtos que ainda não avaliou, com base nas relações descobertas entre os vetores singulares.

As vantagens do **SVD** incluem a capacidade de lidar com dados esparsos, a habilidade de capturar relações complexas e a capacidade de fornecer recomendações precisas. Por essas razões, o **SVD** é uma técnica poderosa para melhorar a personalização e a eficácia dos sistemas de recomendação em e-commerce e em outros contextos onde a análise de dados é fundamental.

4.3.6 Modelo Híbrido

Recapitular que um sistema de recomendações híbrido consiste na unificação de dois ou mais modelos com o único intuito de agregar as diferentes vantagens dos diferentes modelos num só. O modelo híbrido desenvolvido é a junção do sistema de recomendação **FC** ,**FBC** e *Clustering*. As desvantagens de cada modelo são compensadas com as vantagens do próximo:

Modelo	**Vantagens**	**Desvantagens**	**Modelo Mitigante**
FC	- Personalização Dinâmica - Sensibilidade a Novos produtos - Descoberta Social	- Esparsidade do Conjunto de Dados - *Cold Start*	**FBC**, Clustering
FBC	- Recomendações Explicáveis - Robustez a Novos utilizadores - Relevância Contextual	- Falta de Serendipidade - Dependência de Metadados	**FC**, Clustering
Clustering	- Segmentação de Clientes - Adaptação a Mudanças de Comportamento - Redução de Dimensionalidade	- Sensibilidade à Inicialização, Problemas com Outliers	**FC**

Tabela 8: Vantagens, Desvantagens e Modelos Mitigantes no Modelo Híbrido

A tabela 8 apresenta uma análise dos três componentes que constituem o modelo híbrido, destacando as suas vantagens, desvantagens e qual o modelo que promete reduzir as limitações do outro. A **FBC** e

o Modelagem por *clustering*, por exemplo, desempenham papéis cruciais na superação da disparidade do conjunto de dados e na introdução de diversidade, enquanto a **FC** auxilia na adaptação inicial e na sensibilidade à inicialização. Uma explicação mais detalhada encontra-se na secção A.1

Tendo em conta que um dos objetivos é ter um modelo de recomendações facilmente adaptável a todas e qualquer plataforma de negócio, decidiu-se que o peso que cada um dos três modelos tem nos resultados do modelo híbrido pode ser ajustado para maximizar a precisão dos resultados segundo o tipo de plataforma e preferências do utilizador. Por exemplo, se o responsável desejar obter recomendações em que a similaridade entre produtos seja o principal fator nos resultados, é crucial aumentar o peso atribuído à **FBC**, tornando-a o modelo dominante no sistema híbrido.

Com a finalização do modelo e armazenamento dos resultados, cada cliente terá associado um conjunto de produtos que, consoante os três modelos, mais se assemelha às suas preferências.

Capítulo 5

Análise de Resultados

5.1 Avaliação

Neste capítulo, especificam-se as fases finais do processo **CRISP-DM**, avaliação dos modelos. Esta etapa é de extrema importância em qualquer projeto destinado a uso comercial por determinar a eficácia dos resultados obtidos. Caso estes não cumprem os requisitos mínimos ou houver a possibilidade de melhorar, retorna-se à etapa **EDA** para uma reavaliação dos dados e voltar a percorrer o mesmo percurso feito até agora.

5.1.1 Divisão dos dados

Dos dados fornecidos pela transformação de dados, estes vão pertencer a um dos dois conjuntos: o conjunto de dados destinado para treinar os modelos ou o conjunto de dados destinado para avaliar os modelos. Os dados de treino serão utilizados para treinar os algoritmos **ML** previamente definidos e será com base nestes dados que o modelo tentará prever os resultados para o grupo teste.

Para a divisão dos dados utilizou-se a técnica *standard* 80/20, [Caamares et al., 2020] . Uma vez que a ideia é simular um cenário o mais real possível, as encomendas são ordenados da mais antiga à mais recente e 80% dos primeiros valores são utilizados para treinar os modelos, enquanto os restantes 20% são reservados para testar o desempenho dos modelos. Uma vez que o conjunto de teste será constituído pelas encomendas mais recentes presentes no *dataframe*, na prática, as previsões tentam antecipar os pedidos futuros de um cliente, esses "pedidos futuros" estão presentes no conjunto de teste.

5.1.2 Métricas

Esta dissertação sustenta-se na teoria de que a utilização de **IA** em sistemas de recomendação oferece um aumento de vendas às plataformas, isto é, as recomendações feitas a partir do uso de **IA** têm mais sucesso

que recomendações aleatórias ou implementadas manualmente. Apesar de a teoria ser sustentável, esta tem que ser comprovada, é aqui que entram as métricas, estas permitem quantificar a qualidade das recomendações objetivamente.

Precision e Recall

Após uma pesquisa sobre as métricas que mais se enquadram neste projeto, entre as quais [Lü et al., 2012] e [Fayyaz et al., 2020] , tornou-se evidente que as duas métricas *precision* e *recall* são vitais para medir a qualidade dos modelos de recomendação implementados.

A métrica de avaliação **precision** mede a capacidade do sistema de recomendação de sugerir produtos relevantes com base nas preferências do utilizador. A métrica indica a proporção de recomendações que são verdadeiramente úteis para o utilizador, vulgarmente conhecidas como verdadeiros positivos. Uma vez que se quer ter em conta a lista ordenada de produtos mais populares, utiliza-se a variação *precision*@K. *Precision*@K é a métrica que mede a precisão de um sistema de recomendação numa posição específica K na lista de produtos recomendados. Esta métrica indica quantos dos principais K produtos recomendados são relevantes para o utilizador. Se o valor de K for 5, significa que a métrica calcula a proporção de produtos relevantes entre os 5 principais produtos recomendados. Posto isto, se 3 dos 5 principais produtos recomendados forem relevantes, a *precision*@5 será de 3/5 = 0,6 (ou 60%). Valores mais altos de *precision*@K significam que uma maior percentagem dos principais K produtos recomendados são relevantes para o utilizador, indicando uma maior precisão.

$$Precision@K = \frac{\text{Número de produtos relevantes entre os top-k recomendados}}{k}$$

Enquanto *precision* mede a proporção de exemplos classificados como positivos que são verdadeiramente positivos, **recall** mede a capacidade de um modelo identificar todos os exemplos positivos. Em outras palavras, mede quantos dos artigos que realmente importam foram identificados corretamente pelo modelo. Tal como a métrica anterior, foi utilizado a variação **Recall@K**. *Recall*@K é uma métrica que mede a quantidade de produtos relevantes pelo sistema de recomendação numa posição específica K na lista de produtos recomendados. Se o valor de K for 5, significa que a métrica calcula a proporção de produtos relevantes encontrados entre todos os produtos com os quais o utilizador interagiu. Por exemplo, para os 5 principais produtos recomendados, se o utilizador tiver 10 produtos relevantes nos dados de

teste e o sistema recomendar 4 deles nos 5 principais, então o *recall*@5 seria de 4/10 = 0,4 (ou 40%).

$$Recall@K = \frac{\text{Número de produtos relevantes entre os top-k recomendados}}{\text{Número total de produtos relevantes}}$$

MAP

Apesar destas duas métricas apresentarem valores relevantes e capazes de fornecer uma valiosa perspetiva da *performance* dos modelos, nenhuma das duas tem em consideração a ordem pela qual as recomendações são apresentadas, ao contrário da métrica de avaliação **MAP**, [Schröder et al., 2011].

Em termos simples, **MAP** calcula a média das precisões médias para cada utilizador num conjunto de dados de recomendação. No entanto, o que torna esta métrica particularmente valiosa é a consideração da ordem em que os artigos são apresentados nas recomendações. Isto significa que a métrica não se limita a avaliar apenas se um artigo é relevante (*recall*) ou a proporção de produtos relevantes entre os recomendados (*precision*). Este considera a posição relativa dos produtos relevantes na lista de recomendações.

$$MAP = \frac{1}{N}\sum_{k=1}^{N}\frac{P(k)\cdot rel(k)}{k}$$

-N: Número total de utilizadores no conjunto de teste;

- k: Índice que varia de 1 a N, representando a posição na lista de recomendações para cada cliente;

- $p(k)$: precisão nos primeiros k produtos recomendados para o cliente;

- $rel(k)$: função indicadora que verifica se o artigo na posição k é relevante (geralmente definido como 1 para relevante e 0 para não relevante).

Esta métrica apresenta vantagens substanciais relativamente às métricas *precision* e *recall* entre as quais:

- **Sensibilidade à Ordem:** O **MAP** valoriza os sistemas que posicionam os artigos relevantes no topo das recomendações, uma vez que os utilizadores costumam dar mais atenção aos primeiros artigos sugeridos. Este é fundamental, pois o *recall* e a *precision* não consideram essa ordenação.
- **Adequado para Classificação:** O **MAP** é particularmente útil quando o objetivo é classificar artigos por relevância. Ao contrário do *recall* e da *precision*, ele oferece informações sobre a ordem

dos resultados, o que é crucial em sistemas de recomendação.

- **Equilíbrio entre Precisão e Recall:** Esta métrica pondera tanto a qualidade quanto a quantidade das recomendações. Esta característica torna-o flexível para se adaptar aos diferentes objetivos dos sistemas de recomendação.

- **Avaliação Global:** Fornece uma avaliação global do desempenho do sistema, considerando todas as consultas ou utilizadores no conjunto de dados. Isso permite uma visão geral da eficácia do sistema de recomendação.

5.1.3 Variável evaluate

Embora a avaliação das métricas seja importante durante o desenvolvimento dos modelos de recomendação para percebermos se estamos, ou não, a seguir o caminho correto, estas perdem a sua utilidade assim que os modelos, aos olhos do programador, estiverem concluídos.

Tal como se comprova mais adiante nesta dissertação, o tempo que as métricas demoram a "avaliar" ocupa grande parte do tempo de execução e assim que estes modelos estiverem concluídos as percentagens serão sempre as mesmas, o utilizador pode querer as suas recomendações o mais rápido possível e esperar por algo a que já sabemos o valor torna-se obsoleto. Posto isto, foi decidido colocar a avaliação dos modelos como opcional, ou seja, a avaliação será controlada pela variável booleana **evaluate** em que caso esta esteja marcada como "*false*", a avaliação dos modelos será ignorada.

Apesar de ser o tema da dissertação, há que ter em conta que este é um projeto destinado para comércio e será uma ferramenta utilizada várias vezes ao dia por diversas empresas, a ênfase na experiência do utilizador sobrepõe-se à importância relativa das métricas teóricas. A eficácia real do modelo reside na capacidade de oferecer sugestões pertinentes e atrativas, independente das métricas de avaliação. Assim, a demonstração das percentagens de avaliação torna-se menos relevante quando comparada à entrega de recomendações eficientes. A implementação prática da opcionalidade não apenas reduz a complexidade computacional associada ao cálculo das métricas de avaliação, mas também otimiza o desempenho geral do sistema.

5.2 Introdução aos Resultados

Chegou-se ao momento decisivo na nossa jornada, em que será comprovado se todo esforço e dedicação até este momento compensou. Neste capítulo, detalha-se os resultados obtidos na fase de "Avaliação"

do processo **CRISP-DM**.

Os resultados apresentados foram sujeitos a uma análise detalhada, com o intuito de avaliar a verdadeira contribuição da **IA**, especialmente nos sistemas de recomendação, para uma plataforma *e-commerce*. Pretende-se determinar se a implementação desta tecnologia oferece uma vantagem tangível e significativa para a empresa selecionada no âmbito deste estudo.

É importante salientar que todo o processo de análise e avaliação foi conduzido com base em dados reais fornecidos pela DeltaQ Portugal, uma empresa de renome e cliente da Bsolus. Estes dados reais são a base da pesquisa e resultados apresentados, refletindo a autenticidade e relevância necessárias para conduzir uma avaliação sólida e significativa. A expectativa é que os resultados apresentados neste capítulo proporcionem uma visão aprofundada sobre a eficácia da aplicação da **IA** em sistemas de recomendação para um negócio de *e-commerce*, trazendo conclusões valiosas para a empresa envolvida e para a comunidade académica e empresarial.

5.2.1 Interpretação dos resultados obtidos

Os dados fornecidos para obter os resultados da tabela 9 serão do ano 2022 e inícios de 2023, em que os dados de treino serão dados recolhidos entre agosto de 2022 e fevereiro de 2023 e os dados de teste serão desde março até inícios de setembro de 2023. Desta forma será possível criar um cenário em que podemos "prever o futuro". No subcapítulo presente, apresentamos os valores das métricas de desempenho, *precision@K, recall@K* e *MAP@K*.

Modelos	**Precision@5**	**Precision@10**	**Recall@5**	**Recall@10**	**MAP@10**
Base	8.08%	8.33%	7.33%	9.00%	7.00%
Popularidade	12.14%	12.45%	15.65%	32.07%	6.01%
Complementar	10.97%	11.53%	14.14%	26.08%	8.47%
F.Colaborativa	20.45%	23.67%	32.09%	47.89%	30.27%
F. Baseada Conteúdo	18.20%	22.71%	30.25%	41.16%	33.57%
Clustering	18.63%	22.93%	34.06%	44.53%	17.00%
Híbrido	21.99%	25.48%	32.28%	48.33%	35.23%

Tabela 9: Resultados das métricas

Em primeiro lugar, o modelo base mencionado seguiu uma abordagem em que é o cliente o responsável por introduzir manualmente os produtos recomendados, esta introdução não envolveu nenhum tipo

de estrutura ou algoritmo **ML**, apenas uma lógica subjetiva por parte do responsável. Este modelo representa um modelo de recomendação sem qualquer tipo de interação de **IA**. Os resultados evidenciam uma eficácia limitada, com valores de *precision*, *recall* e **MAP** abaixo dos restantes modelos, que envolveram algum tipo de **ML**. Esta abordagem intuitiva levou a recomendações menos relevantes devido à falta de dados estruturados e padrões discerníveis.

O modelo complementar apresenta valores satisfatórios, mas não alcança o patamar dos modelos que compõem o modelo híbrido. Isto acontece principalmente devido à limitação do modelo complementar em considerar apenas produtos comprados em conjunto. A diversidade e imprevisibilidade dos padrões de compra tornam desafiador encontrar relações significativas entre produtos, afetando assim a eficácia do modelo.

Por outro lado, o modelo popularidade destaca-se ao oferecer recomendações satisfatórias para utilizadores que não tiveram interações prévias com a plataforma. A fórmula desenvolvida, que considera a popularidade dos produtos, revelou-se uma estratégia eficaz para sugestões iniciais, alcançando valores adequados de *precision* e *recall*.

Contudo, o ponto alto desta análise reside no modelo híbrido, uma fusão das abordagens de **FC**, **FBC** e *clustering*. Tendo em conta que, para os resultados da tabela 9, foi dado um peso uniforme para os três componentes individuais que o completam, este modelo híbrido mostra melhorias notáveis, especialmente em termos de **MAP**, com aumentos de até 5%. Recordar que **MAP** é uma métrica que avalia a qualidade geral das recomendações, considerando a precisão em diferentes posições do *ranking*. Portanto, um aumento no **MAP** indica uma melhoria na qualidade global das recomendações.

Em conclusão, os resultados desta análise evidenciam que modelos mais simples (popularidade) são superados significativamente por mais complexos (modelo híbrido). Este último, que representa o foco principal desta dissertação e o ápice da **IA** em modelos de recomendação, valida a importância da introdução da **IA** para aprimorar a precisão e relevância das recomendações, destacando-se como a escolha preferencial para sistemas de recomendação mais sofisticados.

Pelos resultados é possível perceber a discrepância entre *recall* e *precision*, uma característica comum nas métricas de avaliação. O *recall*, que mede a capacidade do modelo em capturar todos os produtos relevantes, frequentemente apresenta uma *performance* melhor que a *precision*, que avalia a proporção de produtos recomendados que são realmente relevantes.

Uma razão válida para esta desigualdade entre as duas métricas é a utilização de conjunto de dados "desequilibrados", em que a quantidade de artigos relevantes supera significativamente a de artigos irrelevantes. Os **SR** centram-se apenas nos produtos relevantes, provocando um *recall* mais elevado. No

entanto, esta abordagem pode causar a inclusão de alguns produtos irrelevantes, impactando negativamente na métrica *precision*. Os modelos apresentados, ao priorizar a inclusão de uma ampla gama de produtos relevantes, tendem a abdicar de *precision* para assegurar que nenhum produto relevante seja negligenciado.

Outra razão igualmente válida é a escolha do valor de *K* nas métricas discutidas. O aumento do valor de *K* melhora, geralmente, o *recall*, uma vez que permite a inclusão de mais produtos relevantes, mas pode diminuir a *precision*, pois mais produtos irrelevantes também são considerados.

5.2.2 Configuração Experimental

Híbrido

Uma configuração no modelo Híbrido será diferir o peso atribuído a cada modelo que compõe o modelo híbrido. Para contexto, caso seja atribuído um peso zero a um ou mais dos três modelos envolventes, este deixa de influenciar os resultados do modelo e é como se não existisse para o algoritmo do modelo híbrido. Por exemplo, se for dado um peso de zero aos modelos *clustering* e **FBC**, os resultados do sistema de recomendação híbrido será igual ao resultados do modelo **FC**.

Posto isto, executou-se o sistema com as mesmas configurações utilizadas na recolha das métricas acima listadas, mas foi-se reduzindo o peso alternadamente para determinar qual a melhor configuração possível.

Modelo Híbrido	Precision@10	Recall@10	MAP@10
s/ **FC**	17.76%	36.80%	28.56%
s/ **FBC**	20.27%	38.98%	31.88%
s/ Clustering	23.34%	45.34%	34.76%

Tabela 10: Resultados da configuração experimental no modelo híbrido

Pela tabela acima é facilmente percetível que **FC** é modelo com mais influência no algoritmo híbrido. A razão para qual isto acontece é a "baixa" quantidade de dados treinados. À medida que se aumenta a informação provida ao sistema de recomendações, a probabilidade de resultar numa maior desigualdade nas preferências dos consumidores é bastante alta, o que dificultará o modelo **FC** de encontrar padrões entre clientes e produtos. Por sua vez, o modelo **FBC**, que se baseia apenas nas semelhanças entre produtos, acabará por ter mais influência no modelo final.

Ao passo que o modelo *clustering* foi posteriormente incluído ao sistema de recomendação híbrido,

destaca-se que este tornou-se uma mais-valia para o modelo de recomendações, sendo que quanto menor o seu peso menor a qualidade do sistema. Esta leve melhoria deve-se principalmente à inclusão de clientes com pouco ou nenhum histórico.

Popularidade

Para o modelo de popularidade, a configuração experimental centrou-se na modificação da fórmula previamente mencionada. Inicialmente, foi discutido adicionar as variáveis "*stock*", *created_at* e "*price*" com o intuito de reforçar o *ranking* dos produtos que têm poucas ou nenhumas encomendas e, consequentemente, não se enquadram nas variáveis já selecionadas: *product.quantity*, *product.hits_count* e *status.strength*. No entanto, as variáveis "*stock*" e *created_at* não apresentam um padrão e os seus dados podem ser bastante dispersos, o que provocou um decréscimo na métrica MAP@10 até de 4%. Uma vez que a ideia geral desta abordagem é apresentar até os vinte produtos mais populares da plataforma, produtos com poucas ou nenhumas encomendas nunca irão chegar a este *top*, o que torna irrelevante adicionar estas varáveis à fórmula final.

Quanto à variável *price*, apesar de considerar como um fator importante na decisão de compra de um determinado produto, não se conseguiu chegar a um consenso de como apresentar este campo na fórmula. Para este ser um campo relevante, teria que se considerar um preço baixo como um influenciador negativo para um produto ser popular e, consequentemente, um produto caro como um influenciador positivo ou vice-versa. Uma vez que, na DeltaQ Portugal existe um intervalo considerável de preços (por exemplo, máquina de café e meia dúzia de cápsulas) decidiu-se em equipa não ter em conta esta variável.

Quanto ao peso que cada atributo tem na fórmula final utilizou-se a famosa técnica tentativa-erro. Verificou-se que à medida que se aumenta as percentagens de *product.hits_count* e *product.quantity* a métrica MAP@10 descia consideravelmente. Este acontecimento reafirma a diferente gama de produtos, pois um cliente ao comprar um conjunto de caixas de cápsulas de café e apenas uma máquina de café não significa necessariamente que as cápsulas são procuradas que a máquina de café.

5.2.3 Armazenamento

Inicialmente, realçou-se que cada modelo de recomendações possui um ficheiro próprio dentro da nossa base de dados PostgresDB. O tamanho desse ficheiro é diretamente influenciado pelo número de encomendas, clientes e produtos fornecidos e, à medida que estes números crescem, a quantidade de recomendações geradas também aumenta. Quanto maior o número de dados recolhidos, maior será o volume de dados que será preciso armazenar e processar, resultando em ficheiros de dimensões consi-

deravelmente maiores. Cada algoritmo tem um tipo de recomendação diferente:

Popularidade

Das três possibilidades, este é o modelo mais compacto, em que cada empresa *e-commerce* apenas ocupa uma entrada no *dataframe*. Cada plataforma terá acesso à lista dos seus produtos, ordenados por popularidade.

```
      {
          "id": "45f6ce35-cdef-4fe9-90f8-9ac70047e519" ,
          "tenant_id": "64b507aabdfa298368a9303b",
          "recommendations": ["a1089e00-ec3f-4e58-9183-4a9cd0287013", "
ceb47397-13eb-44d4-b679-5f30e1c14acb", (...) "f76d6546-ae8b-4338-bcb0
-9219505d5628"]
      }
```

Listing 5.1: Amostra da recomendação guardada do modelo Popularidade

Complementar

No modelo Complementar, à esmagadora maioria dos produtos que já possuem um histórico de vendas serão associados produtos que complementem o carrinho do cliente. Cada produto recomendado terá associado um "*support*" que indica, entre 0 e 1, o quanto o produto recomendado complementa o produto original.

```
      {
          "_id": "45f6ce35-cdef-4fe9-90f8-9ac70047e519" ,
          "product_id": "3c41007a-e9d8-4fbf-adcf-a68f3ea7f28d",
          "tenant_id": "64b507aabdfa298368a9303b",
          "recommendations": [
              {
                  "support": 1,
                  "product_id": "318410e6-4a00-4968-a358-7b87dcae5c4e"
              },
              {
                  "support": 0.8,
                  "product_id": "a9ff23af-36c3-4700-956d-85693ae9db22"
              }
          ]
```

```
    }
```

Listing 5.2: Amostra da recomendação guardada do modelo Complementar

Híbrido

Por último, o modelo híbrido, união entre o modelo colaborativo, baseado em conteúdo e *clustering*, terá associado a cada cliente um conjunto de recomendações, ordenadas pelo nível de importância e o peso que os diferentes modelos tiveram na decisão final. O campo "*client_email*" foi adicionado para facilitar a recolha das recomendações para os clientes.

```
    {
        "_id": "45f6ce35-cdef-4fe9-90f8-9ac70047e519" ,
        "client_id": "3c41007a-e9d8-4fbf-adcf-a68f3ea7f28d",
        "client_email": abc@gmail.com
        "tenant_id": "64b507aabdfa298368a9303b",
        "recommendations": [
            [{"score": 1, "scoreCB": 0, "scoreCF": 1, "scoreCL": 1, "
product_id": "fb442e0e-cf85-4e09-8453-3831aac46e1a"}, {"score":
0.8053122934567085, "scoreCB": 0, "scoreCF": 1, "scoreCL":
0.610624586913417, "product_id": "ad5e6582-d801-49aa-83d1-ec4b1a839a96"
}]
        ]
    }
```

Listing 5.3: Amostra da recomendação guardada do modelo Hibrido

5.2.4 Desempenho

O desempenho do sistema de recomendações dependerá diretamente da dimensão dos *dataframes* inicialmente fornecidos. Quanto maior o tamanho dos *dataframes*, maior o tempo de execução, especialmente do tratamento de dados e do modelo híbrido. Os valores obtidos na tabela 11 resultam da execução dos modelos previamente mencionados e contém cerca de 350 mil encomendas, 240 mil clientes e 400 produtos. A implementação de *threads* desempenhou um papel crucial na redução significativa do tempo de execução, passando de 26 minutos e 37 segundos para cerca de 7 minutos e 22 segundos, abarcando desde a preparação inicial até à avaliação dos modelos.

Além disso, ao guardar antecipadamente as métricas de avaliação, conseguimos reduzir o tempo necessário para gerar recomendações para metade do tempo, cerca de 3 minutos e 54 segundos, o mesmo modelo com as mesmas definições pode adicionar ou diminuir entre um minuto de execução. Isto realça a eficiência na gestão do tempo e da adaptabilidade do sistema para responder às necessidades dos clientes de forma rápida e eficaz.

Componentes	**Tempo de execução (minutos)**
Tratamento de Dados	5.32
Popularidade	3.39
Complementar	4.23
Clustering	6.44
F. Colaborativa	16.89
F. Baseada no Conteúdo	7.57
Híbrido	34.77

Tabela 11: Desempenho dos modelos com avaliação

Numa nota mais específica, na tabela 11, é possível consultar o tempo de execução de cada modelo de recomendação. Percetível pela tabela, uma desvantagem que algoritmos **ML** apresentam é a notável diferença de tempo de execução entre modelos altamente treinados (modelo híbrido) e modelos mais simples (modelo popularidade). Esta discrepância no tempo de execução destaca a importância de encontrar um equilíbrio entre a complexidade do modelo, os recursos disponíveis e as expectativas dos utilizadores.

Capítulo 6

Conclusões e trabalho futuro

6.1 Conclusões

Com a evolução das plataformas na nossa sociedade, é seguro afirmar que a inteligência artificial e sistemas de recomendação irão assumir um papel crucial no comércio digital. Como prova, companhias de renome, como *Netflix* e *Amazon*, têm investido consistentemente nesta área, utilizando a mesma para aprimorar a experiência do consumidor e estimular as suas vendas. A inteligência artificial oferece uma experiência mais personalizada e orientada para a individualidade do cliente, aumentando a satisfação do consumidor. Esta abordagem não só melhora a experiência de compra, mas também otimiza a eficiência da plataforma *e-commerce*, uma vez que a oferta de produtos e serviços correspondem à procura do consumidor.

No âmbito da empresa Bsolus, uma empresa que presta serviços de gestão de plataformas *e-commerce*, a incorporação de um sistema de recomendações baseado em inteligência artificial não só coloca a empresa em pé de igualdade concorrentes do mercado, como também a coloca na vanguarda. A ferramenta desenvolvida proporciona aos seus clientes uma forma poderosa para aumentar as vendas e otimizar a experiência do utilizador.

Neste projeto, procedeu-se à leitura dos dados brutos e estabeleceu-se um contrato de dados agnóstico e modelar, que serve como padrão para todas as plataformas que queiram aderir ao modelo de recomendação, assegurando uniformidade na gestão dos dados. Os dados, provenientes do contrato, foram então analisados através da análise exploratória de dados, permitindo um tratamento de dados eficaz e adequado. Como produto final, acessível aos clientes da empresa, encontram-se três modelos de recomendação distintos. Enumerando do mais simples ao mais complexo, o modelo Popular cujo objetivo é destacar os produtos mais populares na plataforma aos clientes, independentemente das suas características ou preferências. O segundo, o modelo complementar, permite sugerir produtos que complementem as compras dos clientes, a ideia deste algoritmo é não ter em conta as características dos

produtos ou dos clientes, mas sim a relação de compra que os produtos têm entre si e a frequência em que os produtos são comprados simultaneamente. O terceiro modelo, o modelo híbrido, combina três abordagens diferentes: filtragem colaborativa, filtragem baseada em conteúdo e *clustering*. Esta abordagem híbrida oferece recomendações detalhadas para o cliente, considerando tanto as preferências do utilizador quanto as características dos produtos, em adição, agrupa os consumidores conforme as suas preferências e características individuais. Após avaliação, através das métricas *recall*, *precision* e MAP, e aperfeiçoamento dos modelos, os resultados são armazenados numa base de dados, prontos para serem utilizados em ações de *marketing* planeadas pelos gestores da plataforma. Concluiu-se que, dos três modelos, apesar de cada um contribuir com finalidades diferentes, o modelo híbrido destacou-se como o mais eficaz, devido à sua capacidade de combinar vantagens das diferentes abordagens e mitigar eventuais desvantagens dos modelos individuais.

A conclusão deste projeto representa um marco significativo no meu percurso e nos estudos sobre a aplicação da inteligência artificial em sistemas de recomendação. Fico satisfeito em afirmar que todos os objetivos propostos foram alcançados com êxito. Este trabalho ampliou substancialmente o meu entendimento no domínio da inteligência artificial, tornando-se uma ferramenta valiosa para o meu futuro como programador.

A avaliação rigorosa dos sistemas de recomendação desenvolvidos nesta dissertação é inequívoca: estes têm o potencial para impulsionar as vendas de uma plataforma *e-commerce*. Os modelos implementados, que incluem o modelo popular, o modelo complementar e o modelo híbrido, demonstraram a sua eficácia ao proporcionar recomendações precisas aos clientes. Essa personalização não só melhora a experiência do consumidor, como também aumenta o lucro das plataformas.

Deste modo, este projeto não só alcançou os objetivos definidos, como também realça a importância da inteligência artificial e dos sistemas de recomendação no contexto do comércio eletrónico. Os sistemas desenvolvidos não só melhoram a eficiência das operações, mas também a satisfação do cliente, um fator crucial para o sucesso de qualquer plataforma.

6.2 Objetivos Cumpridos

Uma vez que os objetivos a cumprir são definidos pela empresa responsável Bsolus, alguns dos objetivos complementares inicialmente delineados foram alterados pelos responsáveis para cumprir as metas posteriormente estabelecidas. No entanto, o objetivo principal deste projeto foi a construção de um modelo de recomendações eficiente, capaz de albergar os dados de qualquer plataforma, selecionada pela em-

presa responsável Bsolus. Com a dissertação concluída e o projeto devidamente testado, este objetivo foi devidamente cumprido. Numa nota mais específica:

1. **Compreensão do fluxo de informação de uma plataforma e-commerce**

   Durante o meu estágio profissional na empresa Bsolus, fui imerso desde o início na compreensão do fluxo de uma plataforma *e-commerce*. Todos os passos do processo, desde a recolha de dados até a geração de recomendações de produtos personalizados para os clientes, foram minuciosamente explicados. Esta jornada proporcionou-me um profundo entendimento de como a plataforma e o mercado funciona, permitindo-me contribuir significativamente para o aprimoramento das recomendações e a satisfação do cliente.

2. **Implementação de um método capaz de albergar qualquer plataforma associada à Bsolus e futuro treino dos respetivos dados**

   Para garantir a compatibilidade com qualquer plataforma que deseje adotar o modelo de recomendações desenvolvido, implementou-se um contrato de dados que estabelece normas "obrigatórias". Este contrato consiste em três conjuntos essenciais de dados: encomendas, clientes e produtos, com colunas predefinidas, exigindo que as empresas forneçam dados específicos. Esta abordagem transformou o sistema numa estrutura completamente modular e flexível o suficiente para acomodar as plataformas associadas à empresa. Este contrato de dados desempenha um papel central neste projeto, sendo fundamental para a aplicação do método **EDA**. O método analítico tornou-se crucial para identificar padrões, relações e anomalias que de outra forma seriam impossíveis de encontrar.

3. **Implementação de novas técnicas de ML, sendo estas as mais aceites e estudadas pela academia até o momento presente;**

   Os modelos de recomendação desenvolvidos aderem rigorosamente aos padrões e normas mais atualizadas, amplamente adotadas por gigantes do setor como a *Amazon*,*Netflix* e *Spotify*. Um exemplo notável de uma nova implementação foi a introdução do modelo complementar, amplamente empregado por empresas de venda de produtos. Isto demonstra o compromisso em manter a relevância e eficácia das recomendações, ao alinhar as estratégias com as melhores práticas do setor.

4. **Análise dos resultados por métricas definidas pela equipa e processos de validação orientados a este tipo de problemas**

Todos os modelos de recomendação implementados passaram por uma avaliação criteriosa. Esta avaliação envolveu a utilização de métricas estrategicamente selecionadas para este tipo de problema, tais como *precision*, *recall* e MAP. Estas métricas foram escolhidas para medir com precisão a eficácia e o desempenho dos modelos, garantindo que as recomendações sejam altamente relevantes e atendam às necessidades específicas do problema de recomendação. A utilização de métricas permite verificar a evolução dos modelos e ter uma ideia de quais são os mais vantajosos a implementar.

5. **Implementação de sistema de recomendações totalmente modular, capaz de abranger qualquer tipo de plataforma e comunicar com a restante ferramenta**

   Como resultado deste projeto temos um sistema de recomendações capaz de comunicar com a restante ferramenta produzida pela Bsolus, em que recebe informação para iniciar de um microserviço adjacente e guarda os resultados numa base de dados independente ao modelo de recomendações. Este serviço torna-se totalmente modular e agnóstico uma vez que, caso cumpram as regras estabelecidas, um *e-commerce* externo pode utilizar esta ferramenta com apenas alguns passos e sem qualquer tipo de intervenção de um responsável para modificar esta ferramenta para que o *e-commerce* externo receba resultados. No entanto, este responsável pode tentar modificar as configurações para a ferramenta servir o melhor possível a plataforma externa.

6.3 Perspetiva de trabalho futuro

Com o projeto atual concluído, a empresa Bsolus dispõe de um conjunto de modelos de recomendações alinhados com os objetivos definidos para este projeto, capaz de fornecer recomendações fidedignas aos clientes, aprimorando assim a experiência do cliente na plataforma *e-commerce*. No entanto, é importante realçar que este sucesso atual não é o fim da jornada, mas sim o ponto de partida para um processo constante de evolução e melhoramento. Conscientes de que há sempre margem para aperfeiçoamento, pretende-se não só introduzir novos modelos de recomendação, mas também explorar diferentes abordagens da utilização de inteligência artificial em plataformas *e-commerce*.

Uma das prioridades a adicionar ao nosso modelo de recomendações será a possibilidade de plataformas **B2B** e **B2E** poderem utilizar a nossa ferramenta de recomendações. Apesar de a ferramenta estar acessível e funcional a diferentes tipos de comércios, a Bsolus cogita criar uma ferramenta adicional de recomendações totalmente direcionada para empresas que desejam implementar sistemas de recomendação personalizados, seja para aprimorar a experiência das suas empresas **B2B** ou facilitar a

tomada de decisões internas **B2E**. Esta expansão da ferramenta permitirá expandir o alcance e o impacto da ferramenta desenvolvida, permitindo a entrada de todo o tipo de empresas para a nossa plataforma.

Quanto à avaliação dos modelos, um ponto extremamente interessante seria receber um *feedback* explícito dos utilizadores no que toca às recomendações feitas a si. Para tal, a ideia é incentivar ativamente o envolvimento dos utilizadores, permitindo que expressem as suas preferências e forneçam contribuições valiosas que orientarão ajustes específicos nos modelos. O inconveniente deste "*upgrade*" é que exige a interação de uma quantia considerável de utilizadores/clientes e conseguir processar esta informação de forma autónoma. Apesar de já existir uma avaliação "contínua" do desempenho dos modelos a partir das métricas previamente mencionadas, a equipa responsável tem com objetivo futuro implementar o método testes A/B para avaliar o impacto direto de alterações específicas, refinando constantemente os nossos modelos em busca de melhores resultados. Este objetivo não está marcado como urgente, mas fará certamente a diferença na eficiência dos modelos.

Relativamente aos modelos já desenvolvidos, pretende-se no futuro ter em conta a variável "promoções", principalmente no modelo Popularidade. Após um breve estudo, torna-se óbvio que a redução do preço de um produto pode torná-lo muito mais desejável a um cliente, melhorando assim a sua "popularidade" numa plataforma. Num mundo ideal, cada produto terá um atributo promoção (entre 0% e 100%) em que o seu valor será tido em conta no seu *score*.

Fora dos sistemas de recomendação, no futuro pretende-se introduzir *newsletters* personalizadas para as plataformas *e-commerce*. As *newsletters* são mensagens periódicas enviadas por *e-mail* aos clientes, apresentando novos produtos, ofertas especiais, atualizações e conteúdo relevante. Este meio de comunicação direta e personalizada pretende manter os clientes informados, fortalecendo o relacionamento com a plataforma.

Outra inovação em construção é a criação de um *chatbot* capaz de interagir diretamente com os utilizadores da plataforma. Este assistente virtual, baseado em inteligência artificial, proporcionará um suporte personalizado aos clientes, respondendo às suas perguntas e poderá auxiliar na seleção de um ou mais produtos, com base nas suas preferências e necessidades sem a necessidade de um responsável intervir. Em termos ligeiros pretende-se oferecer uma experiência interativa e amigável aos clientes, tal como se estivesse numa loja real.

Por fim, um objetivo mais distante, mas igualmente interessante, é a aplicação da inteligência artificial na previsão de demanda e gestão de *stocks*. Neste problema recorrente em múltiplas empresas, serão utilizados algoritmos **ML** para prever com precisão as necessidades de *stock*, capaz de evitar tanta a falta de produtos quanto o excesso de *stock*. Com a introdução do histórico de vendas, condições climáticas e

até tendências no mercado atual em algoritmos **ML** possibilitará uma ferramenta às plataformas cliente capaz de prever a oferta/procura de *stock* com base nos padrões de compra, comportamento dos seus clientes e tendências de mercado.

Esta constante necessidade de querer melhorar a ferramenta de inteligência artificial permitirá à empresa manter-se na vanguarda do mercado *e-commerce*, não só satisfazendo as expectativas atuais, mas também antecipando as futuras necessidades dos consumidores, proporcionando uma experiência de compra única e à medida.

Bibliografia

Pegah Malekpour Alamdari, Nima Jafari Navimipour, Mehdi Hosseinzadeh, Ali Asghar Safaei, and Aso Darwesh. A systematic study on the recommender systems in the e-commerce. *Ieee Access*, 8: 115694–115716, 2020.

Gonçalo Almeida. Crisp-dm is still the most popular framework for executing data science projects, 2022. URL https://bsolusdev.atlassian.net/wiki/spaces/MA/pages/225181715/Sphinx+Documentation. Acessado em 20 de Novembro de 2023.

Amazon. Amazon sagemaler, 2023. URL https://aws.amazon.com/sagemaker/. Acessado em 20 de Novembro de 2023.

G Anev. The nature and the scope of the modern e-commerce. *International Independent Scientific Journal*, (29):13–16, 2021.

J. Konstan B. Sarwar, G. Karypis and J. Riedl. *"Item-based collaborative filtering recommen- dation algorithms",*. Proceedings of ACM World Wide Web Conference„ 2001.

Roco Caamares, Pablo Castells, and Alistair Moffat. Offline evaluation options for recommender systems. *Information Retrieval Journal*, 23(4):387–410, 2020.

José Costa. Data mining: Modelação de algoritmos para automação de marketing. *University of Minho*, 1:43–49, 2021.

Tommaso Di Noia, Nava Tintarev, Panagiota Fatourou, and Markus Schedl. Recommender systems under european ai regulations. *Communications of the ACM*, 65(4):69–73, 2022.

Vaishnavi Nath Dornadula and Sa Geetha. Credit card fraud detection using machine learning algorithms. *Procedia computer science*, 165:631–641, 2019.

Zeshan Fayyaz, Mahsa Ebrahimian, Dina Nawara, Ahmed Ibrahim, and Rasha Kashef. Recommendation systems: Algorithms, challenges, metrics, and business opportunities. *applied sciences*, 10(21):7748, 2020.

José Costa Gonçalo Almeida. Engine documentation, 2022. URL https://bsolusdev.atlassian.net/wiki/spaces/MA/pages/15433729/MA+Data+Engine. Acessado em 20 de Novembro de 2023.

Jiawei Han, Jian Pei, and Hanghang Tong. *Data mining: Concepts and Techniques*. Morgan kaufmann, 2022.

Liaoliang Jiang, Yuting Cheng, Li Yang, Jing Li, Hongyang Yan, and Xiaoqin Wang. A trust-based collaborative filtering algorithm for e-commerce recommendation system. *Journal of ambient intelligence and humanized computing*, 10:3023–3034, 2019.

Ramez Kian. Provide a model for an e-commerce system with the impact of artificial intelligence. *International Journal of Innovation in Management, Economics and Social Sciences*, 1(3):88–94, 2021.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

Linyuan Lü, Matúš Medo, Chi Ho Yeung, Yi-Cheng Zhang, Zi-Ke Zhang, and Tao Zhou. Recommender systems. *Physics reports*, 519(1):1–49, 2012.

M. Malik M. Hasan, S. Ahmed and S. Ahmed. *"A comprehensive approach towards user-based collaborative filtering recommender system",*. International Journal of Scientific Technology Research,, 2016.

Yaser Ahangari Nanehkaran. An introduction to electronic commerce. *International journal of scientific & technology research*, 2(4):190–193, 2013.

Nils J Nilsson. *Artificial intelligence: a new synthesis*. Morgan Kaufmann, 1998.

Michael J Pazzani and Daniel Billsus. Content-based recommendation systems. In *The adaptive web: methods and strategies of web personalization*, pages 325–341. Springer, 2007.

Jeff Saltz. Crisp-dm is still the most popular framework for executing data science projects, 2022. URL https://www.datascience-pm.com/crisp-dm-still-most-popular/. Acessado em 20 de Novembro de 2023.

Gunnar Schröder, Maik Thiele, and Wolfgang Lehner. Setting goals and choosing metrics for recommender system evaluations. In *UCERSTI2 workshop at the 5th ACM conference on recommender systems, Chicago, USA*, volume 23, page 53, 2011.

Christoph Schröer, Felix Kruse, and Jorge Marx Gómez. A systematic literature review on applying crisp-dm process model. *Procedia Computer Science*, 181:526–534, 2021.

Lalita Sharma and Anju Gera. A survey of recommendation system: Research challenges. *International Journal of Engineering Trends and Technology (IJETT)*, 4(5):1989–1992, 2013.

Brent Smith and Greg Linden. Two decades of recommender systems at amazon. com. *Ieee internet computing*, 21(3):12–18, 2017.

Ms K Susmitha. Impact of covid 19 on e-commerce. *Journal of Interdisciplinary Cycle Research*, 12(9): 1161–1165, 2021.

Yan Tian and Concetta Stewart. History of e-commerce. In *Electronic commerce: concepts, methodologies, tools, and applications*, pages 1–8. IGI Global, 2008.

John W Tukey et al. *Exploratory data analysis*, volume 2. Reading, MA, 1977.

Robin Van Meteren and Maarten Van Someren. Using content-based filtering for recommendation. In *Proceedings of the machine learning in the new information age: MLnet/ECML2000 workshop*, volume 30, pages 47–56. Barcelona, 2000.

Kanit Wongsuphasawat, Yang Liu, and Jeffrey Heer. Goals, process, and challenges of exploratory data analysis: An interview study. *arXiv preprint arXiv:1911.00568*, 2019.

Qian Zhang, Jie Lu, and Yaochu Jin. Artificial intelligence in recommender systems. *Complex & Intelligent Systems*, 7:439–457, 2021.

Parte I
Apêndices

Apêndice A
Detalhes dos resultados

Detalhes de resultados cuja extensão comprometeria a legibilidade do texto principal.

A.1 Explicação detalhada vantagens e desvantagens dos componentes do modelo híbrido

1. **Filtragem Colaborativa**

   Vantagens:

   - Personalização Dinâmica: Captura padrões de preferências em constante evolução.
   - Sensibilidade a Novos produtos: Pode recomendar produtos com base no histórico de utilizadores com gostos semelhantes, mesmo que sejam novos para o utilizador atual.
   - Descoberta Social: Explora preferências comuns entre utilizadores, promovendo a descoberta de produtos por meio da rede social do cliente.

   Desvantagens:

   - Esparsidade do Conjunto de Dados: Enfrenta dificuldades quando as interações são limitadas.
   - Frio Inicial (Cold Start): Enfrenta desafios ao recomendar a novos utilizadores sem histórico de interações.

   Mitigação:

   - Filtragem Baseada em Conteúdo: Supre a esparsidade, fornecendo recomendações para utilizadores com históricos limitados.
   - Modelagem por Clustering: Ajuda na adaptação inicial e supera a esparsidade do conjunto de dados.

2. **Filtragem Baseada em Conteúdo**

   Vantagens:

   - Recomendações Explicáveis: Baseia-se nas características intrínsecas dos produtos, tornando as recomendações mais explicáveis e interpretáveis.
   - Robustez a Novos Utilizadores: Não depende do histórico de outros utilizadores, sendo útil para recomendar a novos utilizadores ou utilizadores com históricos limitados.
   - Relevância Contextual: Considera os atributos dos produtos, alinhando melhor as recomendações com as preferências específicas do utilizador.

   Desvantagens:

   - Falta de Serendipidade: Tende a recomendar produtos semelhantes.
   - Dependência de Metadados: Requer dados detalhados sobre os atributos dos produtos.

   Mitigação:

   - Filtragem Colaborativa: Introduz diversidade ao sugerir produtos inesperados.
   - Modelagem por Clustering: Ajuda na segmentação de produtos mais diversificada. </enditemize

3. **Modelagem por Clustering**

   Vantagens:

   - Segmentação de Clientes: Agrupa utilizadores com características semelhantes, permitindo recomendações mais precisas dentro de grupos específicos.
   - Adaptação a Mudanças de Comportamento: Se ajusta a mudanças nas preferências dos utilizadores ao refazer periodicamente os clusters.
   - Redução de Dimensionalidade: Pode lidar com grandes conjuntos de dados e reduzir a complexidade ao agrupar utilizadores em clusters significativos.

   Desvantagens:

   - Sensibilidade à Inicialização, Problemas com Outliers: A formação inicial de clusters pode ser sensível à escolha de parâmetros e qualidade dos dados.

Mitigação:

- Filtragem Colaborativa: Ajuda a superar a sensibilidade à inicialização, fornecendo insights sobre preferências individuais.

Apêndice B
Listings

B.1 Exemplificação de uma função teste

```
@pytest.mark.calculateAge
@mock.patch("rs_eda_dp.calculateAge")

def test_calculateAge(calculate_age_mock: mock.Mock):
    """function to test method calculateAge

    :param string_to_int_mock: Replicated calculateAge function using @mock
    .patch
    :type string_to_int_mock: mock.Mock

    :return: assert between the result dataframe and the expected dataframe
    """

    df = [{ "created_at": "2020-10-22 16:18:29"}]

    dfRes = [{"product.age": 3}]
    dfRes = pd.DataFrame(dfRes)

    calculate_age_mock.return_value = pd.DataFrame(df)

    results = calculateAge(
        calculate_age_mock.return_value, "created_at", "product.age", today
    )

    results = pd.DataFrame(results)
    assert_frame_equal(results, dfRes, check_dtype=False)
```

Listing B.1: Exemplificação de uma função teste

Coloque aqui informação sobre financiamento, projeto FCT, etc. em que o trabalho se enquadra. Deixe em branco caso contrário.