



Sistema de Recomendação em tempo real para E-commerce

RICARDO JORGE COELHO LIMA

Outubro de 2020



Sistema de Recomendação em tempo real para E-commerce

Redicom

2019 / 2020

1130499 - Ricardo Jorge Coelho Lima

Sistema de Recomendação em tempo real para E-commerce

Redicom

2019 / 2020

1130499 - Ricardo Jorge Coelho Lima



Mestrado em Engenharia de Software

14 de Outubro de 2020

Orientador ISEP: **Carlos Ferreira**

Supervisor Externo: **Serafim Costa**

Agradecimentos

Sem a ajuda e apoio de determinadas pessoas esta dissertação nunca se poderia realizar e, com isto, quero deixar aqui, por escrito, o meu especial e carinhoso agradecimento às seguintes pessoas:

- Primeiro gostaria de agradecer ao meu orientador, o Professor Doutor Carlos Ferreira, por toda a orientação, atenção e disponibilidade dada. Sem ele esta dissertação nunca se teria concretizado e teria sido muito complicado encontrar o caminho correto a seguir. Ao Professor, o meu muito obrigado por nunca me ter deixado sair do caminho certo e por me instruir acerca de todos os temas desta dissertação;
- Gostaria também de deixar aqui a minha apreciação ao meu supervisor, Serafim Costa, pela disponibilidade, auxílio e confiança para que o projeto se concretizasse. Ao meu supervisor, o muito obrigado;
- Não poderia deixar de agradecer aos meus pais por fazerem de mim o que sou, por inculcaram em mim os ideais que defendo e por estarem sempre presentes, quando mais precisava, com uma palavra e gesto amigo;
- Nunca me poderia esquecer da minha ímpar “mulher” que nunca me deixou ir abaixo. Que teve presente durante todo o processo para me ajudar, apoiar, criticar quando assim o entendia, mas acima de tudo, me amar e acarinhar para que esta jornada fosse a mais cómoda e familiar possível. Sem ela nada disto seria possível e desde já quero deixar o meu especial obrigado por tudo aquilo que fez por mim e para mim. O meu especial e amoroso obrigado;
- Aos meus sogros e irmão da minha namorada por estarem presentes nos melhores momentos da minha vida e me darem a mão e ajudarem nos momentos menos fáceis;
- Aos avós da minha namorada por estarem presentes todos os dias e me ajudarem em todos os momentos. Sem nenhuma obrigação familiar para comigo, acolheram-me e cuidam de mim como se fosse família e isso, nunca esquecerei.

Por último, peço desde já desculpa caso me tenha esquecido de alguém, mas deixo desde já o meu muito obrigado a todos aqueles que me ajudaram e acompanharam, direta, ou indiretamente, durante todo este processo.

Resumo

Nos dias de hoje, o E-commerce tem tido uma crescente adesão tanto por parte das empresas como dos utilizadores, o que leva conseqüentemente a um aumento significativo no interesse por sistemas de recomendação.

Devido ao facto de as empresas conseguirem obter cada vez mais informações, é necessário a utilização de um sistema capaz de processar as mesmas. Esses sistemas de recomendação fazem assim uso de toda a informação com a finalidade de recomendarem os produtos e serviços aos utilizadores.

É na ideia do desenvolvimento de um sistema de recomendação que esta dissertação se assenta. Serão descritas técnicas e métodos capazes de processar a informação, em conjunto com as suas vantagens e limitações. Serão também analisadas várias métricas capazes de avaliar as recomendações dadas por um determinado sistema.

Serão realizadas várias avaliações a diferentes métodos de recomendação utilizando, diferentes métricas como avaliação, tendo como base vários *datasets* de treino, com diferentes características.

Com todas essas avaliações e análises dos diferentes métodos, esta dissertação culminou na implementação de um sistema de recomendação, sobre a forma de *Web Service*, que utiliza um mecanismo de recomendações baseado em regras associativas e nos produtos mais populares. Com esta solução é possível utilizar de uma forma otimizada a informação obtidas acerca dos utilizadores tornando, simultaneamente, este um processo automático para os clientes.

Palavras-chave: Sistema de recomendação, E-commerce, *Web Service*.

Abstract

On the present day, E-commerce has an increasing adhesion both by companies and users, which leads to a significant increase in interest on recommendation systems.

Since companies can obtain more and more information, it is necessary to use a system that is capable of processing that information. These recommendation systems thus make use of all information for the purpose of recommending products and services to users.

Is on the idea of developing a recommendation system that this dissertation is based on. Techniques and methods capable of processing information will be described, alongside with its advantages and limitations. Various metrics capable of evaluating the recommendations given by the system will also be analysed.

Several evaluations of recommendations methods will be carried out using different metrics, based on several training datasets, with different characteristics.

With all these evaluations and analysis of the different methods, this dissertation culminated on the implementation of a recommendation system, as a *Web Service*, which uses a recommendation mechanism based on associative rules and the most popular products. With this solution, it is possible to use the collected information about users in an optimized way, making this process an automatic process for customers.

Índice

| | | |
|----------|--|-----------|
| 1 | <i>Introdução</i> | 1 |
| 1.1 | Contexto | 1 |
| 1.2 | Problema | 1 |
| 1.3 | Objetivos | 2 |
| 1.4 | Estrutura do relatório | 3 |
| 2 | <i>Análise de valor</i> | 5 |
| 2.1 | Análise de valor segundo o modelo NCD | 5 |
| 2.2 | Análise funcional | 7 |
| 2.3 | Avaliação e seleção das tecnologias a utilizar | 10 |
| 2.3.1 | Cálculo e análise da prioridade relativa de cada critério..... | 11 |
| 2.3.2 | Construção da matriz de comparação de cada alternativa..... | 13 |
| 2.3.3 | Seleção da tecnologia..... | 14 |
| 3 | <i>Estado da Arte</i> | 16 |
| 3.1 | Sistemas de recomendação | 16 |
| 3.1.1 | Recolha de informação para o sistema de recomendação..... | 17 |
| 3.2 | Técnicas utilizadas em sistemas de recomendação | 19 |
| 3.2.1 | Filtragem baseada no conteúdo..... | 20 |
| 3.2.2 | Filtragem colaborativa..... | 21 |
| 3.3 | Métodos de avaliação da qualidade de sistemas de recomendação | 25 |
| 3.3.1 | Estudos de Utilizadores..... | 25 |
| 3.3.2 | Avaliação <i>offline</i> | 25 |
| 3.3.3 | Avaliação <i>online</i> | 26 |
| 3.3.4 | Métricas de avaliação da qualidade do sistema de recomendação..... | 26 |
| 3.4 | Aplicações reais de sistemas de recomendação | 29 |
| 3.4.1 | Trabalhos relacionados..... | 29 |
| 4 | <i>Propostas de soluções</i> | 31 |
| 4.1 | Análise de requisitos | 31 |
| 4.2 | Apresentação das soluções | 32 |
| 4.2.1 | Primeira Solução..... | 32 |
| 4.2.2 | Segunda Solução..... | 33 |

| | | |
|------------|---|-----------|
| 4.2.3 | Terceira solução..... | 34 |
| 5 | Casos de estudo | 36 |
| 5.1 | Análise dos utilizadores..... | 36 |
| 5.2 | Avaliação dos métodos de recomendação | 39 |
| 5.2.1 | Identificação das amostras a utilizar | 40 |
| 5.2.2 | Métodos de recomendação..... | 43 |
| 5.2.3 | Processo de avaliação dos métodos de recomendação | 46 |
| 5.2.4 | Resultados da avaliação para AR | 48 |
| 5.2.5 | Resultados da avaliação para IBCF | 51 |
| 5.2.6 | Resultados da avaliação para UBCF | 55 |
| 5.2.7 | Resultados da avaliação de todos os métodos de recomendação | 58 |
| 6 | Implementação da solução..... | 62 |
| 6.1 | Método de recomendação | 62 |
| 6.2 | Arquitetura da solução..... | 63 |
| 6.2.1 | Camada Services | 65 |
| 6.2.2 | Camada <i>Models</i> | 66 |
| 6.2.3 | Camada DAL..... | 66 |
| 6.3 | Funcionalidades da solução..... | 67 |
| 6.3.1 | Inserir compras no sistema..... | 67 |
| 6.3.2 | Treinar o modelo | 69 |
| 6.3.3 | Identificar recomendações | 70 |
| 6.4 | Implantação da solução..... | 71 |
| 6.5 | Requisitos satisfeitos | 72 |
| 7 | Conclusão | 74 |
| 7.1 | Objetivos cumpridos | 74 |
| 7.2 | Trabalhos futuros..... | 75 |

Índice de figuras

| | |
|--|----|
| Figura 1 – Esquematização do modelo NCD..... | 5 |
| Figura 2 - Diagrama FAST adaptado ao presente projeto | 8 |
| Figura 3 - Associação entre funcionalidades e entidades | 9 |
| Figura 4 - Árvore hierárquica de decisão..... | 11 |
| Figura 5 - Esquematização dos valores a utilizar para a seleção da tecnologia | 14 |
| Figura 6 - Esquematização das fases de um sistema de recomendação..... | 17 |
| Figura 7 - Ilustração de categorias dos sistemas de recomendação | 19 |
| Figura 8 - Esquematização da hierarquia das técnicas..... | 20 |
| Figura 9 - Representação de uma vizinhança composta por três vizinhos | 23 |
| Figura 10 - Representação do espaço ROC e esquematização da análise das curvas..... | 28 |
| Figura 11 - Diagrama de implantação da primeira solução..... | 33 |
| Figura 12 - Diagrama de implantação da segunda solução..... | 33 |
| Figura 13 - Diagrama de implantação da terceira solução..... | 34 |
| Figura 14 - Processo de análise para selecionar o melhor método de recomendação | 40 |
| Figura 15 - Excerto de uma matriz utilizada para treinar o sistema de recomendação | 41 |
| Figura 16 - Matriz dos dados da primeira amostra | 42 |
| Figura 17 - Matriz dos dados da segunda amostra | 42 |
| Figura 18 - Matriz dos dados da terceira amostra | 43 |
| Figura 19 - Exemplo de uma lista de métodos a utilizar para recomendação | 44 |
| Figura 20 - Esquema do método de filtragem colaborativa baseada em utilizadores..... | 45 |
| Figura 21 - Esquema do método de filtragem colaborativa baseada em produtos..... | 45 |
| Figura 22 - Esquema de avaliação cruzada..... | 46 |
| Figura 23 - Configuração utilizada para as avaliações..... | 46 |
| Figura 24 - Exemplo da configuração para avaliação dos algoritmos | 47 |

| | |
|--|----|
| Figura 25 - Código para o mecanismo de recomendação de produtos aos utilizadores | 63 |
| Figura 26 - Diagrama de componentes da solução | 64 |
| Figura 27 - Diagrama de pacotes da solução..... | 65 |
| Figura 28 - Código para a disponibilização dos serviços da solução | 66 |
| Figura 29 - Exemplo do corpo de um pedido para adicionar compras com duas compras | 68 |
| Figura 30 - Diagrama de sequência para a funcionalidade de inserir compras no sistema | 69 |
| Figura 31 - Diagrama de sequência para a funcionalidade de treinar o modelo | 70 |
| Figura 32 - Diagrama de sequência para a funcionalidade de identificar recomendações | 71 |
| Figura 33 - Diferença entre Docker e virtualização | 72 |
| Figura 34 - Pedido cURL para inserir compras no sistema | 84 |

Índice de tabelas

| | |
|---|----|
| Tabela 1 - Análise da relevância de cada funcionalidade para o sistema | 10 |
| Tabela 2 - Matriz de comparação entre os critérios da hierarquia | 12 |
| Tabela 3 - Cálculo da prioridade relativa de cada critério..... | 12 |
| Tabela 4 - Valores do IR para matrizes quadradas de ordem n | 13 |
| Tabela 5 - Matriz de comparação para a capacidade para identificar recomendações | 13 |
| Tabela 6 - Matriz de comparação para a maturidade | 14 |
| Tabela 7 - Matriz de comparação para a modularidade | 14 |
| Tabela 8 - Descrição dos requisitos funcionais e não funcionais | 32 |
| Tabela 9 - Resumo das características de cada amostra..... | 40 |
| Tabela 10 - Configurações mais relevantes para cada amostra do método AR..... | 51 |
| Tabela 11 - Configurações mais relevantes para cada amostra do método IBCF | 54 |
| Tabela 12 - Configurações mais relevantes para cada amostra do método UBCF | 58 |
| Tabela 13 - Configurações que melhores recomendações realizaram, em cada amostra..... | 59 |
| Tabela 14 - Descrição das funcionalidades disponibilizadas pelo Web Service | 67 |
| Tabela 15 – Tempo necessário para inserir um conjunto de compras na base de dados | 68 |
| Tabela 16 – Escala fundamental de Saaty com os níveis de importância de comparações.... | 82 |
| Tabela 17 - Resultados das avaliações ao método AR para a primeira amostra..... | 87 |
| Tabela 18 - Resultados das avaliações ao método AR para a segunda amostra..... | 88 |
| Tabela 19 - Resultados das avaliações ao método AR para a terceira amostra..... | 89 |
| Tabela 20 - Resultados das avaliações ao método IBCF para a primeira amostra..... | 90 |
| Tabela 21 - Resultados das avaliações ao método IBCF para a segunda amostra | 91 |
| Tabela 22 - Resultados das avaliações ao método IBCF para a terceira amostra | 92 |
| Tabela 23 - Resultados das avaliações ao método UBCF para a primeira amostra | 93 |
| Tabela 24 - Resultados das avaliações ao método UBCF para a segunda amostra | 94 |

| | |
|--|----|
| Tabela 25 - Resultados das avaliações ao método UBCF para a terceira amostra..... | 95 |
| Tabela 26 - Resultados das avaliações a todos os métodos para a primeira amostra..... | 96 |
| Tabela 27 - Resultados das avaliações a todos os métodos para a segunda amostra | 97 |
| Tabela 28 - Resultados das avaliações a todos os métodos para a terceira amostra | 98 |

Índice de gráficos

| | |
|--|----|
| Gráfico 1 - Número de produtos comprados em cada família | 37 |
| Gráfico 2 - Percentagem de produtos comprados em cada género | 37 |
| Gráfico 3 - Valores gastos em produtos para cada género | 38 |
| Gráfico 4 - Relação entre o valor médio gasto por encomenda e o número de dias até à próxima compra..... | 38 |
| Gráfico 5 – Valores da F-Measure (Primeira amostra – AR)..... | 48 |
| Gráfico 6 - Curva ROC (Primeira amostra - AR) | 49 |
| Gráfico 7 – Valores da F-Measure (Segunda amostra – AR) | 49 |
| Gráfico 8 - Curva ROC (Segunda amostra - AR) | 50 |
| Gráfico 9 – Valores da F-Measure (Terceira amostra – AR) | 50 |
| Gráfico 10 – Valores da F-Measure (Primeira amostra - IBCF)..... | 52 |
| Gráfico 11 - Curva ROC (Primeira amostra - IBCF)..... | 52 |
| Gráfico 12 - Valores da F-Measure (Segunda amostra - IBCF) | 53 |
| Gráfico 13 - Valores da F-Measure (Terceira amostra - IBCF) | 53 |
| Gráfico 14 - Curva ROC (Terceira amostra - IBCF) | 54 |
| Gráfico 15 - Valores da F-Measure (Primeira amostra - UBCF)..... | 55 |
| Gráfico 16 - Curva ROC (Primeira amostra - UBCF) | 56 |
| Gráfico 17 - Valores da F-Measure (Segunda amostra - UBCF)..... | 56 |
| Gráfico 18 - Curva ROC (Segunda amostra - UBCF)..... | 57 |
| Gráfico 19 - Valores da F-Measure (Terceira amostra - UBCF) | 57 |
| Gráfico 20 - Curva ROC (Terceira amostra - UBCF)..... | 58 |
| Gráfico 21 - Valores da F-Measure (Primeira amostra – todos os métodos)..... | 59 |
| Gráfico 22 - Valores da F-Measure (Segunda amostra – todos os métodos)..... | 60 |
| Gráfico 23 - Valores da F-Measure (Terceira amostra – todos os métodos) | 60 |

Notação e Glossário

| | |
|-------------|--|
| AHP | Processo analítico hierárquico (<i>Analytic Hierarchy Process</i>) |
| B2B | Negócio-para-Negócio, ou <i>Business-to-Business</i> , trata-se do comércio entre duas empresas para que o produto seja vendido ao consumidor final, por exemplo, a empresa que produz o produto e a empresa retalhista que, por sua vez, esta última irá levar o produto ao consumidor final. |
| B2C | Negócio-para-Consumidor, ou <i>Business-to-Consumer</i> , trata-se do comércio direto entre o negócio e o consumidor final |
| CARS | <i>Context-Aware Recommendation System</i> – Um sistema de recomendação com percepção do contexto |
| FAST | <i>Técnica para análise de funcionalidades do sistema (Function Analysis System Technique)</i> |
| ML | <i>Aprendizagem automática (Machine Learning)</i> |
| NCD | <i>New Concept Development</i> |
| OS | Sistema Operativo, <i>Operating System</i> |
| RS | Sistema de Recomendação |
| VM | Máquina Virtual, <i>Virtual Machine</i> |

1 Introdução

Neste capítulo será feita uma introdução à dissertação que se realiza no âmbito do Mestrado em Engenharia de Software do Instituto Superior de Engenharia do Porto, no ano letivo de 2019/2020. Serão abordados e descritos temas como o problema, as motivações e os objetivos deste projeto.

1.1 Contexto

O conceito de E-commerce diz respeito às atividades comerciais realizadas através de meios eletrónicos. Por outras palavras, trata-se da transformação de transações tradicionais em transações realizadas através de meios eletrónicos (Qin et al., 2009).

A Redicom foi fundada em 2002 com o objetivo de desenvolver soluções inovadoras para o comércio eletrónico em associação a várias marcas nacionais e internacionais, tendo assim uma forte presença no mercado de E-commerce. Noutros termos, a Redicom disponibiliza, aos seus clientes¹, soluções que lhes proporcionem uma presença no mercado digital de forma a apresentarem os seus produtos a utilizadores², bem como gerirem todas as particularidades do seu negócio, tais como stocks, preços, faturação, entre outras.

De entre as soluções disponibilizadas pela Redicom, evidencia-se uma plataforma de E-commerce, ou, por outras palavras, uma loja online (B2C, ou B2B) para o negócio da empresa em conjunto com um sistema para gestão de todo o seu conteúdo, *Content Management System* (CMS) e uma plataforma de gestão de transações, *Order Management System* (OMS), para gestão de todas as transações feitas pelos utilizadores da solução.

1.2 Problema

Atualmente, uma das funcionalidades disponibilizadas pela plataforma de E-commerce é a recomendação de produtos, aos utilizadores, na página de detalhe de um produto. Estas recomendações são configuradas manualmente no CMS pelos clientes, o que, além de extremamente moroso, também é propenso a erros (ENACHE, 2019).

¹ Empresas que contratam os serviços da Redicom.

² Clientes das empresas que contratam os serviços da Redicom.

Além da forma como as recomendações são configuradas, o número de produtos que o utilizador tem à sua disposição pode também ser um problema, pelo que é necessário a utilização de qualquer tipo de filtro para que a informação não seja avassaladora e, por consequência, faça com que a experiência de compra, para os utilizadores, seja negativa (Isinkaye et al., 2015).

Foi com estes problemas que se observou a oportunidade para a implementação de um sistema de recomendação automático de produtos.

Por sistema de recomendação entende-se um conjunto de ferramentas e técnicas utilizadas com a finalidade de apresentar, aos utilizadores, recomendações e sugestões que lhes sejam úteis no processo de decisão sobre qual produto comprar (Raza & Ding, 2019).

Devido ao desenvolvimento e expansão do E-commerce, os sistemas de recomendação têm atraído grande interesse. Recentemente, dado o facto de as empresas conseguirem obter grandes quantidades de informações, tais como, informação demográfica dos seus clientes, fluxo de cliques nas plataformas e informação relativa a transações, é possível que os sistemas de recomendação façam uso destas informações com a finalidade de recomendarem os produtos e serviços, de entre várias opções, mais adequados a cada utilizador (So & Yada, 2017).

Em conjunto, os sistemas de recomendação podem ser considerados sistemas de filtragem que lidam com o problema da sobrecarga da informação, apresentado poucos produtos aos utilizadores de entre uma grande variedade existente, ajudando assim o utilizador no processo de decisão (Isinkaye et al., 2015; Patel & Dharwa, 2016).

A adoção de sistemas de recomendação traz assim benefícios para o negócio pois a ajuda dada aos utilizadores para que consigam encontrar, de uma forma mais rápida, os produtos que mais gostam leva a um maior número de transações, bem como a um aumento da satisfação do utilizador no ato da compra (ENACHE, 2019).

1.3 Objetivos

A presente dissertação tem como principal objetivo a implementação de um sistema de recomendação que, a partir do perfil do utilizador, dos produtos existentes na plataforma e das experiências de cada utilizador na plataforma, tais como: produtos vistos, produtos comprados, produtos clicados, entre outras, seja capaz de obter as seguintes informações:

1. Produtos mais prováveis de um utilizador comprar;

2. Segmentação dos utilizadores;
3. Ordenação dos produtos, pelos mais prováveis de serem comprados, por cada segmento identificado.

Os objetivos podem-se dividir em duas partes: Gerais e Específicos. Os objetivos gerais dizem respeito aos objetivos do ponto de vista do negócio, enquanto que os objetivos específicos se referem ao ponto de vista da implementação da solução.

No que diz respeito aos objetivos gerais, consideram-se os seguintes:

- Aumento do número de transações na plataforma utilizando as recomendações dadas pela solução implementada;
- Aumento do número de cliques nos produtos recomendados pela solução implementada.

Como objetivos específicos, têm-se os seguintes:

- Implementação de um sistema de recomendação capaz de obter as informações pretendidas, a partir das experiências dos utilizadores;
- Automatização de processos de forma a otimizar processos no cliente.

1.4 Estrutura do relatório

A presente dissertação divide-se em quatro capítulos principais, respetivamente: introdução, análise de valor, estado da arte e propostas de soluções.

No primeiro capítulo será abordado o problema que originou este projeto, tal como as motivações que levaram à sua elaboração. Serão ainda analisados os objetivos que são pretendidos atingir com a conclusão deste projeto.

De seguida, no capítulo da análise de valor, será feito um enquadramento do problema no modelo NCD. Posteriormente realizar-se-á uma análise funcional do projeto onde serão evidenciadas as principais funcionalidades, através da elaboração de um diagrama FAST. Após a identificação das funcionalidades, serão descritas as principais entidades interessadas na conclusão do projeto e será enquadrado de que forma cada entidade se relaciona com cada funcionalidade. Por último será utilizado o método AHP para fazer uma avaliação e escolha da tecnologia a utilizar para a implementação da lógica do sistema de recomendação.

No estado da arte, serão analisados os conceitos base dos sistemas de recomendação, desde a obtenção de informação até à avaliação das recomendações identificadas. De seguida serão

analisados alguns exemplos de sistemas de recomendação atualmente utilizados por empresas, bem como outros trabalhos relacionados com a área do problema.

Por último, serão descritos os requisitos do sistema e, em conjunto com toda a informação examinada e obtida nos capítulos anteriores, serão apresentadas três arquiteturas possíveis para a solução final. Serão discutidas as características comuns e diferentes entre elas, bem como a referência e comparação entre tecnologias que possam ser utilizadas em determinados componentes delas.

Será, ainda, elaborado um capítulo relativo às avaliações dos métodos passíveis de serem utilizados na solução final de forma a deliberar qual aquele que apresenta os melhores resultados. Para isso será feito um estudo acerca do comportamento de compras de alguns utilizadores da plataforma, bem como, serão apresentadas e descritas as amostras e metodologia a utilizar nas avaliações e os respetivos resultados e análise dos mesmos.

A partir de toda esta informação obtida, descrita e analisada nos capítulos anteriores, será feito um capítulo onde se irá detalhar a solução final, desde a arquitetura da mesma, até às escolhas tecnológicas, funcionalidades disponibilizadas e requisitos satisfeitos.

Por último, nas conclusões, será feita uma reflexão de todo o trabalho feito anteriormente e quais os objetivos que este cumpriu, bem como uma enumeração de todo o trabalho futuro que beneficiaria este projeto.

2 Análise de valor

Neste capítulo será abordada a perspectiva do negócio face ao trabalho a realizar e encontra-se dividida em três partes. Numa primeira parte será feita uma análise com base no *New Concept Development Model* (NCD), na segunda parte será identificada e feita a análise funcional do trabalho a realizar e, por fim, na terceira parte serão comparadas diferentes tecnologias de forma a avaliar e selecionar aquela que mais se adequa para o desenvolvimento da solução final.

2.1 Análise de valor segundo o modelo NCD

Como referido anteriormente, neste capítulo será feita uma análise baseando-se no modelo NCD descrito em Koen et al., (2001).

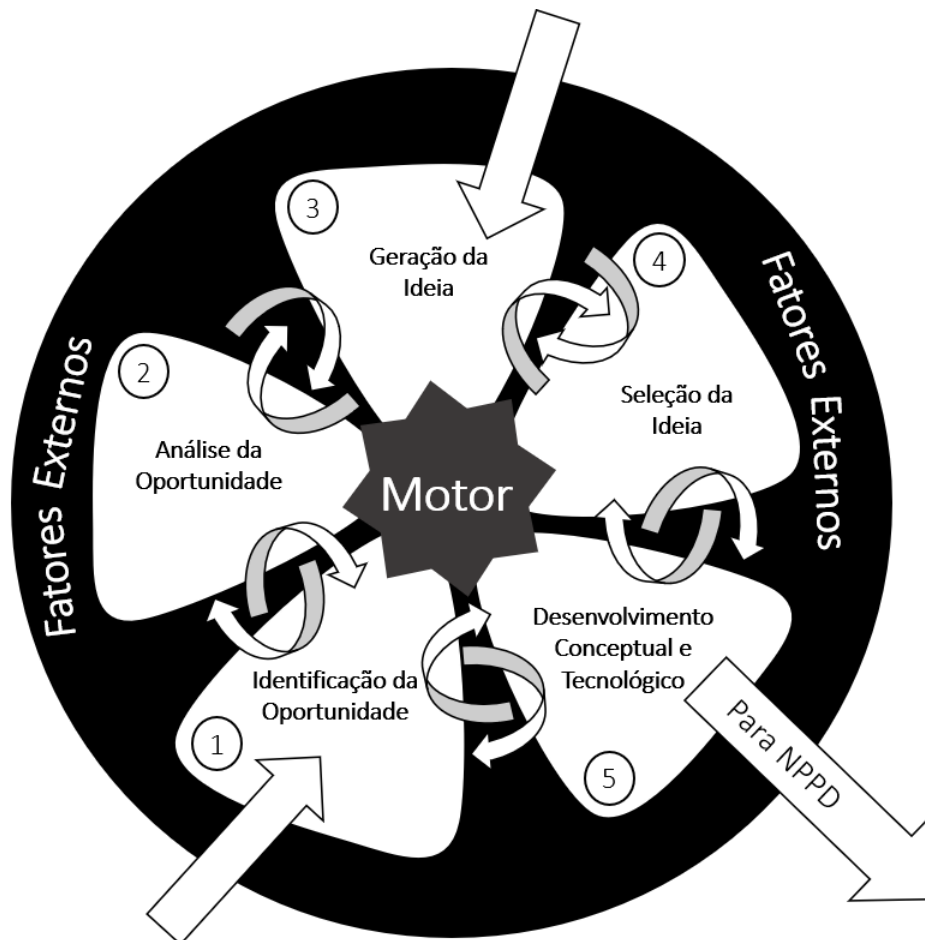


Figura 1 – Esquematização do modelo NCD³

³ Fonte: (Koen et al., 2001)

Na figura anterior podem ser identificados três partes, sendo estas: (Koen et al., 2001):

- Os cinco elementos chave que compõem o modelo:
 1. **Identificação da oportunidade** – Normalmente, originadas pelos objetivos da própria empresa, são identificadas oportunidades de negócio. Estas oportunidades podem-se traduzir em vantagens competitivas, atualização de produtos já existentes, ou otimizações de processos internos;
 2. **Análise da oportunidade** – De forma a tomar a melhor decisão sobre que oportunidades identificadas se tornarão oportunidades concretas da empresa, é necessária a obtenção de informação sobre as mesmas. Essa obtenção pode ser feita através de estudos de mercado, ou experiências científicas;
 3. **Geração da ideia** – A geração da ideia nada mais é que a origem, desenvolvimento e amadurecimento da oportunidade identificada e analisada anteriormente numa ideia real;
 4. **Seleção da ideia** – Com a geração de várias ideias, é necessário, através de processos simples, ou formais, proceder à seleção da ideia que mais beneficie o negócio;
 5. **Desenvolvimento conceptual e tecnológico** – Pode ser equiparado ao desenvolvimento de um plano de negócio, onde é abordado o grau de viabilidade dos processos, o custo associado ao marketing, requisitos técnicos, entre outros aspetos que estão relacionados com o projeto.
- O elemento denominado de “Motor”, responsável por acionar e guiar os cinco elementos mencionados no ponto anterior é impulsionado pela liderança e cultura organizacional;
- Os fatores externos (que consistem nas capacidades organizacionais, estratégias de negócio, canais de distribuição, empresas concorrentes, entre outros), que representam o ambiente e os fatores influenciadores que, por consequência, afetam as restantes partes que compõem o modelo.

No capítulo 1.2 encontra-se descrito o problema que deu origem ao projeto. De uma forma sucinta, os clientes configuram as recomendações para cada produto, individualmente, de uma forma manual. Isto faz com que a atribuição das recomendações seja feita seguindo o critério que o cliente entender que seja o melhor e seja um processo moroso de se fazer.

Adaptando o modelo NCD ao problema associado a este projeto, surge então uma oportunidade de tornar este processo mais eficiente e personalizado para os utilizadores, bem como menos moroso para os clientes.

De acordo com a oportunidade identificada, foi mostrado num estudo realizado pela Infosys, (2013) que “59% dos utilizadores que experienciaram algum tipo de personalização acreditam que isso foi um fator notório que influenciou a sua compra” e que “todos aqueles que oferecem alguma personalização experienciaram pelo menos um benefício na adoção desta tecnologia, sendo os mais citados: aumento das vendas (74%), aumento do lucro (61%), aumento do tráfego online (58%) e aumento da fidelização dos clientes (55%), sendo todos citados pela maioria”.

Gerou-se, a partir da análise anterior, a ideia de recomendar produtos aos clientes de uma forma automática e personalizada. Como a ideia gerada, para este contexto, foi apenas uma, esta encontra-se selecionada por defeito.

A partir da ideia selecionada, será então analisado e definido um plano para que culmina no desenvolvimento de um sistema capaz de recomendar produtos a utilizadores baseando-se, para isso, nas experiências passadas dos utilizadores na plataforma. Com a execução deste plano, a ideia prova-se eficaz para o aproveitamento e cumprimento da oportunidade identificada no primeiro passo, ou seja, que o processo interno de configurar as recomendações para cada produto seja mais eficaz e menos morosa para os clientes.

2.2 Análise funcional

Técnica para análise de funcionalidades do sistema, *Function Analysis System Technique* (FAST), originou-se por volta dos anos sessenta como resposta à evolução que os produtos estavam a sofrer relativamente ao seu tamanho e complexidade (Bartolomei & Miller, 2001; DRM Associates, 2016).

Diagramas FAST podem ser um elemento bastante descritivo das funcionalidades do sistema, bem como do pensamento crítico sobre as mesmas, mesmo que as pessoas que o estejam a analisar não tenham os mesmos contextos técnicos (DRM Associates, 2016).

A análise do diagrama pode ser iniciada da direita para a esquerda, começando com uma funcionalidade e seguindo essa direção através da questão de como essa funcionalidade é alcançável. Outra análise pode ser feita da esquerda para a direita, questionando a finalidade da funcionalidade (DRM Associates, 2016).

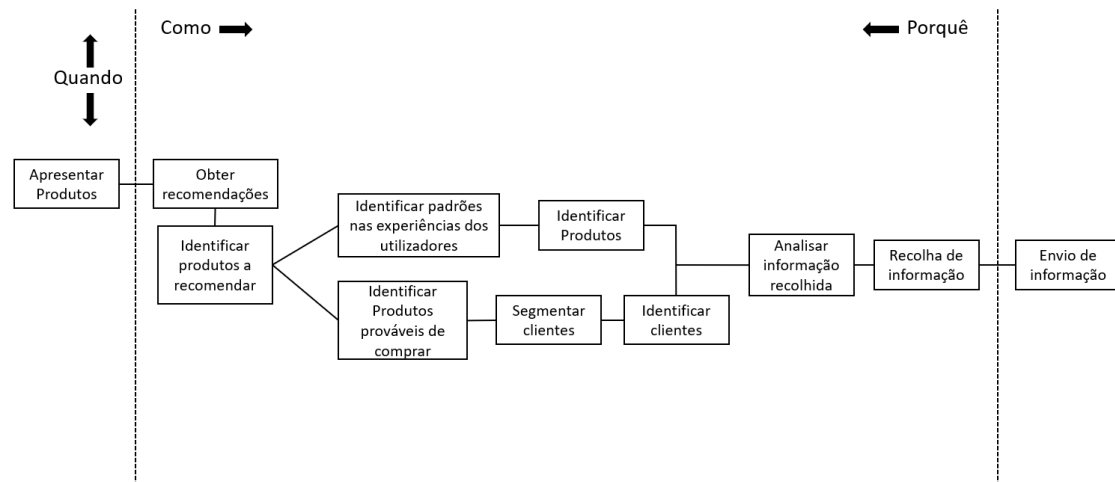


Figura 2 - Diagrama FAST adaptado ao presente projeto

Com base no diagrama apresentado, podem ser identificadas as seguintes funcionalidades:

- Recolher informação;
- Segmentar clientes;
- Identificar os produtos mais prováveis de um segmento comprar;
- Identificar padrões nas experiências dos utilizadores.

Além das funcionalidades acima identificadas, é possível também, identificar três entidades com interesse nas mesmas:

- **Empresa** – Empresa que disponibiliza soluções de comércio eletrónico, de entre as quais, o presente projeto (Redicom);
- **Cientes** – Empresas que contratam os serviços da Redicom;
- **Utilizadores** – Clientes das empresas que contratam os serviços da Redicom.

Com isto, as funcionalidades podem ser associadas às entidades e, a partir dessa associação, atribuir-lhes um grau de importância. Para isso, encontra-se esquematizado na seguinte figura as associações existentes:

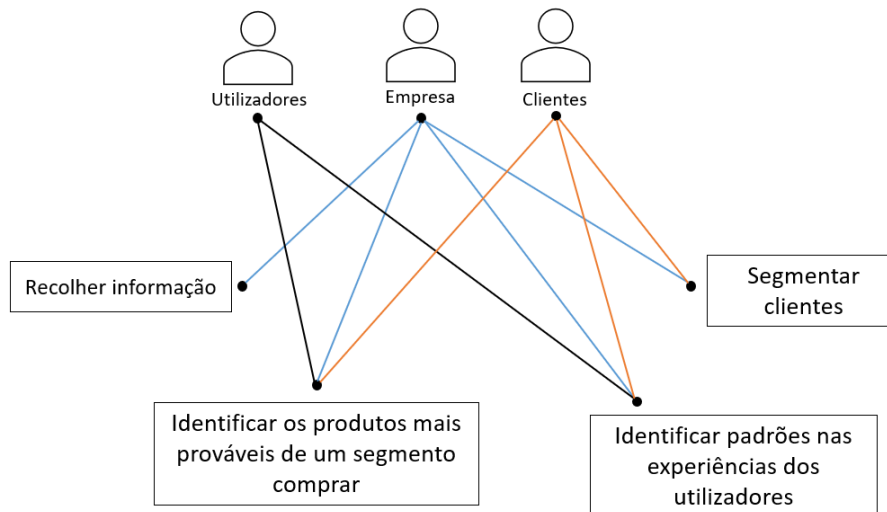


Figura 3 - Associação entre funcionalidades e entidades

Da figura anterior podem ser observadas várias associações. Para a empresa, todas as funcionalidades são de interesse. A identificação de padrões nas experiências e a identificação de produtos mais prováveis de um segmento comprar são fundamentais para a identificação das recomendações, tendo assim uma solução que acrescenta valor aos negócios dos clientes. A recolha de informação e a segmentação de clientes são também importantes, mas não da mesma forma, pois além de serem funcionalidades a utilizar pela empresa no contexto deste projeto, podem também, posteriormente, ser utilizadas num outro contexto. Por exemplo, posicionar um utilizador num determinado segmento além de ser uma informação necessária para o atual projeto, pode ser uma característica a utilizar pela empresa de forma a posicionar os utilizadores para tomadas de decisão futuras.

Tendo em consideração os clientes, tanto a identificação de padrões nas experiências como a identificação de produtos mais prováveis de um segmento comprar, apresentam um valor acrescido para o seu negócio, pois permitem que seja dada aos utilizadores um tipo de personalização no ato de compra e, como foi evidenciado no estudo mencionado no capítulo anterior, “59% dos utilizadores que experienciaram algum tipo de personalização acreditam que isso foi um fator notório que influenciou a sua compra” e que “todos aqueles que oferecem alguma personalização experienciaram pelo menos um benefício na adoção desta tecnologia” (Infosys, 2013). Além disso, a aglomeração de clientes é, da mesma forma que para a empresa, uma funcionalidade importante para os clientes.

Por último, apenas as funcionalidades de identificação de padrões nas experiências e de identificação de produtos mais prováveis de um segmento comprar é que lhes acrescentam valor e personalização à experiência, sendo que as restantes não são conhecidas para eles.

Com base na análise anterior, cada funcionalidade pode ser comparada com outra de forma a verificar qual a que, de entre as duas, é a mais relevante para o sistema. Aquela que, no final obtiver o somatório maior é considerada a mais relevante para o sistema.

| | Recolher informação | Segmentar clientes | Identificar produtos mais prováveis de um segmento comprar | Identificar padrões nas experiências dos utilizadores | Relevância |
|---|---------------------|--------------------|--|---|------------|
| Recolher informação | | 1 | 1 | 0 | 2 |
| Segmentar clientes | 0 | | 1 | 0 | 1 |
| Identificar os produtos mais prováveis de um segmento comprar | 0 | 0 | | 0 | 0 |
| Identificar padrões nas experiências dos utilizadores | 1 | 1 | 1 | | 3 |

Tabela 1 - Análise da relevância de cada funcionalidade para o sistema

Da tabela anterior pode-se retirar a seguinte lista ordenada de funcionalidades segundo o seu grau de relevância para o sistema:

1. Identificar padrões nas experiências dos utilizadores;
2. Recolher informação;
3. Segmentar clientes;
4. Identificar os produtos mais prováveis de um segmento comprar.

2.3 Avaliação e seleção das tecnologias a utilizar

Neste capítulo será utilizado o método de análise de processo analítico hierárquico, *Analytic Hierarchy Process (AHP)*, de forma a auxiliar na tomada de decisão acerca da tecnologia a ser utilizada no presente projeto (Saaty, 1980).

O único requisito levantado para a seleção das tecnologias a serem comparadas é o facto de serem de código aberto (*open-source*) e face esse requisito, foram selecionadas as seguintes tecnologias:

- **TensorFlow** – Plataforma de *Machine Learning* que facilita a criação e implantação de Modelos (*Why TensorFlow | TensorFlow*);
- **Recommenderlab (R)** – Biblioteca que disponibiliza técnicas de *Machine Learning*, bem como a implementação de algoritmos próprios;
- **PredictionIO®** - Plataforma de *Machine Learning* que facilita a construção e implantação de mecanismos de *Machine Learning* como um *WebServices* capazes de responder a pedidos em tempo real (*Welcome to Apache PredictionIO®!*).

Estas tecnologias terão assim de ser comparadas entre si através dos seguintes critérios identificados através da análise feita nos capítulos anteriores:

- **Capacidade para implementar um sistema** – Curva de aprendizagem necessária para que seja implementado um mecanismo capaz de recomendar produtos;
- **Maturidade** – Nível em que a tecnologia se encontra relativamente à estabilidade das funcionalidades que disponibiliza face ao tempo em que a mesma se encontra disponível para utilização;
- **Modularidade** – Capacidade que a tecnologia tem para implementar novos mecanismos ou substituir mecanismos já existentes.

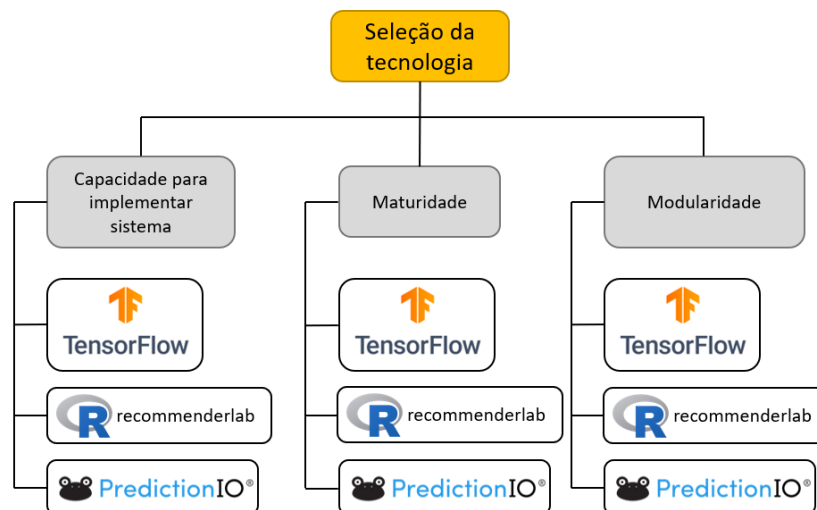


Figura 4 - Árvore hierárquica de decisão

2.3.1 Cálculo e análise da prioridade relativa de cada critério

A partir da árvore definida na figura anterior será necessária a definição do grau de importância entre cada funcionalidade. Com isto, cada funcionalidade será comparada com as restantes para que lhes seja atribuído um valor de acordo com a tabela existente no Anexo A - Escala fundamental de Saaty.

| | | | |
|--|--|-------------------|---------------------|
| | Capacidade para implementar sistema | Maturidade | Modularidade |
| Capacidade para implementar sistema | 1 | 3 | 4 |
| Maturidade | $1/3$ | 1 | 2 |
| Modularidade | $1/4$ | $1/2$ | 1 |

Tabela 2 - Matriz de comparação entre os critérios da hierarquia

De seguida, a matriz anterior será normalizada de forma a que, a partir do cálculo da média aritmética de cada um dos valores normalizados, seja identificada a prioridade relativa, ou vetor próprio de cada critério.

| | | | | |
|--|--|-------------------|---------------------|----------------------------|
| | Capacidade para implementar sistema | Maturidade | Modularidade | Prioridade relativa |
| Capacidade para implementar sistema | $12/19$ | $2/3$ | $4/7$ | 0,62 |
| Maturidade | $4/19$ | $2/9$ | $2/7$ | 0,24 |
| Modularidade | $3/19$ | $1/9$ | $1/7$ | 0,14 |

Tabela 3 - Cálculo da prioridade relativa de cada critério

Após a obtenção da prioridade relativa de cada critério tem de se verificar a confiança dos julgamentos realizados anteriormente. Para isso, será calculada a razão de consistência (RC) e, caso esta seja menor que 0,1, ou seja, 10%, então significa que os julgamentos são confiáveis. Caso o valor da RC obtido seja demasiado alto, então significa que os julgamentos são emparelhados, pelo que o processo terá de ser revisto até que os julgamentos sejam confiáveis.

Para o cálculo da RC é preciso, primeiro, obter o índice de consistência (IC), dado pela fórmula (1), bem como o índice aleatório (IR).

$$IC = \frac{\lambda_{max} - n}{n - 1} \quad (1)$$

Onde λ_{max} é o maior valor próprio da matriz de comparação e n a ordem da matriz.

O valor do IR é obtido através da correspondência do valor referente à ordem da matriz dado na Tabela 4.

| | | | | | | | | | | | | | | | |
|----|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| N | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| IR | 0,00 | 0,00 | 0,58 | 0,90 | 1,12 | 1,24 | 1,32 | 1,41 | 1,45 | 1,49 | 1,51 | 1,48 | 1,56 | 1,57 | 1,59 |

Tabela 4 - Valores do IR para matrizes quadradas de ordem n

Estes dois índices são então utilizados para definir a fórmula da RC da seguinte forma:

$$RC = \frac{IC}{IR} \quad (2)$$

A partir dos cálculos visíveis no Anexo B - Cálculo do valor de RC, foi possível obter uma RC igual a 0,02, o que faz com que os julgamentos tidos anteriormente sejam confiáveis. Com isto, o próximo passo passa pela obtenção das matrizes de comparação para cada uma das alternativas indicadas de forma a que, no final, seja selecionada a que melhor se adequa à resolução do problema.

2.3.2 Construção da matriz de comparação de cada alternativa

Nesta fase, será considerada cada tecnologia na construção das matrizes de comparação para cada critério.

2.3.2.1 Capacidade para implementar um sistema

A avaliação desta característica será baseada nas características que cada tecnologia oferece e também na diversidade da mesma aquando da implementação de um sistema de aplicações. Além dos fatores mencionados anteriormente, será também considerada a documentação e exemplos que cada tecnologia disponibiliza para suportar e encaminhar a implementação de sistemas de recomendação.

| | TensorFlow | Recommenderlab | PredictionIO® | Vetor próprio |
|----------------|------------|----------------|---------------|---------------|
| TensorFlow | 1 | 1/3 | 1/2 | 0,16 |
| Recommenderlab | 3 | 1 | 4 | 0,62 |
| PredictionIO® | 2 | 1/4 | 1 | 0,22 |

Tabela 5 - Matriz de comparação para a capacidade para identificar recomendações

2.3.2.2 Maturidade

Dado que as tecnologias são de código aberto, para a avaliação deste critério ter-se-á em consideração o ano em que a tecnologia foi disponibilizada, bem como a comunidade que contribui para o crescimento do projeto e a quantidade contribuições que cada uma obteve ao longo desses anos. A partir dessa análise será obtido um grau de maturidade entre elas.

| | TensorFlow | Recommenderlab | PredictionIO® | Vetor próprio |
|----------------|---------------|----------------|---------------|---------------|
| TensorFlow | 1 | 5 | 4 | 0,68 |
| Recommenderlab | $\frac{1}{5}$ | 1 | $\frac{1}{2}$ | 0,12 |
| PredictionIO® | $\frac{1}{4}$ | 2 | 1 | 0,20 |

Tabela 6 - Matriz de comparação para a maturidade

2.3.2.3 Modularidade

Para a avaliação da modularidade que cada tecnologia disponibiliza, serão consideradas funcionalidades que estes disponibilizem que contribuam para esta característica.

| | TensorFlow | Recommenderlab | PredictionIO® | Vetor próprio |
|----------------|---------------|----------------|---------------|---------------|
| TensorFlow | 1 | 4 | $\frac{1}{2}$ | 0,33 |
| Recommenderlab | $\frac{1}{4}$ | 1 | $\frac{1}{5}$ | 0,10 |
| PredictionIO® | 2 | 5 | 1 | 0,57 |

Tabela 7 - Matriz de comparação para a modularidade

2.3.3 Seleção da tecnologia

Para a execução da última fase, será calculada a prioridade composta a partir da matriz de comparação construída no capítulo 2.3.2 e representada na Figura 5.

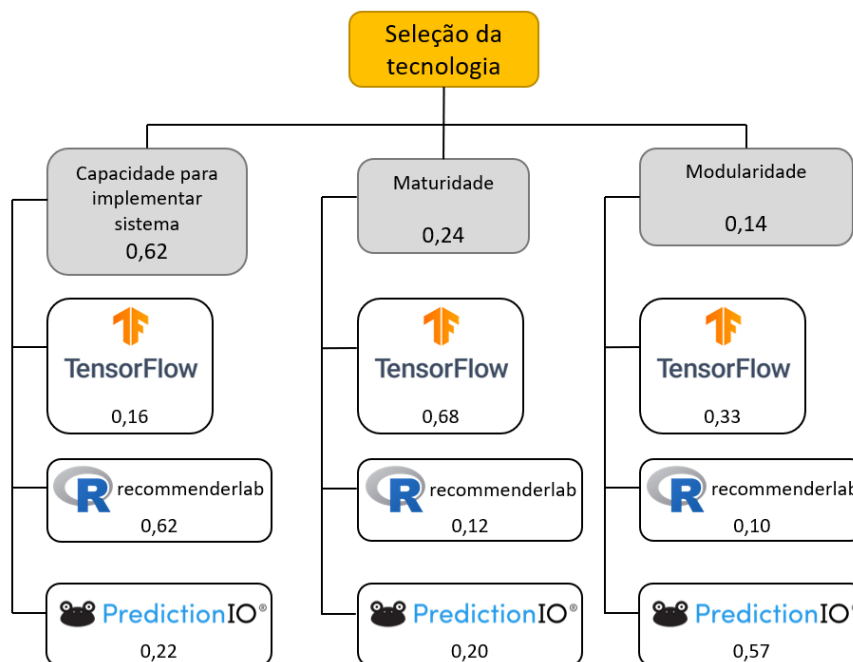


Figura 5 - Esquematização dos valores a utilizar para a seleção da tecnologia

Para isso, será multiplicada a matriz com as prioridades relativas de cada critério com a matriz composta pelos vetores próprios calculados nos capítulos anteriores.

$$\begin{bmatrix} 0,16 & 0,62 & 0,22 \\ 0,68 & 0,12 & 0,20 \\ 0,33 & 0,10 & 0,57 \end{bmatrix} \times \begin{bmatrix} 0,62 \\ 0,24 \\ 0,14 \end{bmatrix} = \begin{bmatrix} 0,31 \\ \mathbf{0,43} \\ 0,27 \end{bmatrix}$$

A partir dos cálculos evidenciados, foi identificada, mediante os critérios utilizados, que a tecnologia mais para que utilização é o recommenderlab.

3 Estado da Arte

No presente capítulo será realizada uma análise detalhada e mais aprofundada sobre os sistemas de recomendação. Inicialmente será escrutinado o conceito de sistemas de recomendação bem como as noções base afetas aos mesmos.

De seguida serão abordados conceitos acerca da informação necessária para o funcionamento, seguido das diferentes técnicas de filtragem utilizada nos sistemas de recomendação e terminado nos métodos de avaliação destes sistemas.

Por fim serão referidos e analisados alguns exemplos sobre a forma como as empresas utilizam de forma benéfica os sistemas de recomendação, assim como outros trabalhos relacionados com estes sistemas.

3.1 Sistemas de recomendação

Dado o crescimento do E-commerce, um dos principais desafios passa por ajudar os utilizadores a identificarem, de entre uma grande variedade de produtos oferecidos, aqueles que mais se adequam às suas preferências. Uma das ferramentas capaz de enfrentar e ultrapassar este desafio são os sistemas de recomendação (Hu et al., 2008).

Sistemas de recomendação são sistemas de aprendizagem automática que ajudam o utilizador a encontrar, ou descobrir, os produtos que são mais do seu gosto através da análise da sua informação e comportamento (ENACHE, 2019; Hwangbo et al., 2018).

Tradicionalmente, as abordagens às recomendações são baseadas em duas dimensões (utilizadores \times produtos) onde um conjunto de N produtos são previstos para serem do interesse de M utilizadores (Adomavicius & Tuzhilin, 2005). Para lidar com esta situação, a comunidade, nos anos 2000, começou a investigar a noção de contexto nas abordagens às recomendações (Raza & Ding, 2019). Aqui originou-se um novo sistema de recomendação conhecido como *Context-Aware Recommender System* (CARS) (Adomavicius & Tuzhilin, 2005). No CARS, as duas dimensões tradicionais são estendidas para potenciar a informação contextual de forma a que sejam sugeridas melhores recomendações personalizadas aos utilizadores (Raza & Ding, 2019).

O processo de recomendação de produtos, como esquematizado na Figura 6, é normalmente constituído por três fases: fase de recolha de informação, fase de aprendizagem e fase de recomendação (Isinkaye et al., 2015; Kumar, 2019).

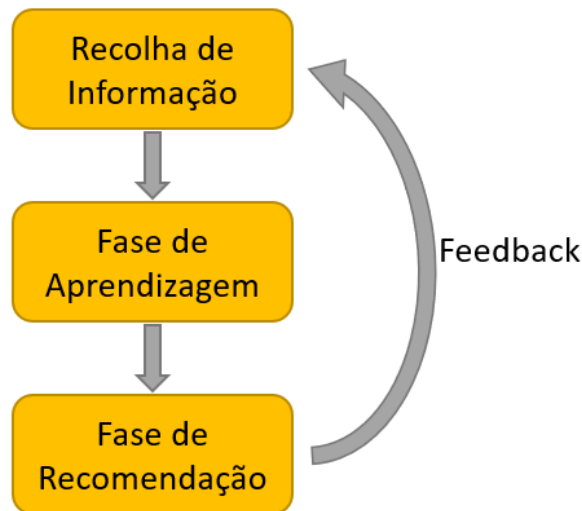


Figura 6 - Esquematização das fases de um sistema de recomendação

A primeira fase centra-se na obtenção de informação que seja considerada relevante para as fases seguintes. Nesta fase são recolhidas informações relativas às características dos utilizadores e produtos, bem como informações comportamentais sobre os utilizadores, tais como: produtos vistos, produtos comprados, entre outras (Isinkaye et al., 2015).

A fase seguinte implica a utilização de um algoritmo de aprendizagem sobre a informação obtida na fase anterior de forma a que a informação consiga ser explorada em função do algoritmo utilizado (Isinkaye et al., 2015).

Por último, a fase de recomendação diz respeito ao ato de recomendar um produto que seja da preferência do utilizador, baseando-se, para isso, na informação explorada na fase anterior (Isinkaye et al., 2015).

Mediante o feedback obtido às recomendações realizadas, quer seja através de avaliações ao sistema, quer seja feedback através de um inquérito aos utilizadores acerca das recomendações que lhes foram mostradas, o sistema de recomendação pode voltar à sua primeira fase para que as recomendações realizadas sejam mais precisas e mais prováveis de o utilizador comprar (Isinkaye et al., 2015).

3.1.1 Recolha de informação para o sistema de recomendação

A primeira fase caracteriza-se pela obtenção de informação necessária para as restantes fases do sistema. Esta informação divide-se em dois tipos: explícita e implícita (Hu et al., 2008; Isinkaye et al., 2015).

Informação explícita consiste em avaliações diretas ao produto realizadas por utilizadores. Por exemplo, quando, depois de uma compra, é pedido ao utilizador para avaliar o produto

mediante uma determinada escala. Esta avaliação dada pelo utilizador enquadra-se no âmbito de informação explícita. Por outro lado, informação implícita consiste na informação relativa às experiências dos utilizadores: produtos comprados, produtos vistos, produtos clicados, entre outras (Herlocker et al., 2004).

Como dito anteriormente, um dos requisitos para o sistema é o facto de este ter de se basear nas experiências dos utilizadores para fazer as recomendações. Estas experiências, como a partir da definição anterior, identificam-se como informação implícita, sendo que, para este caso, é importante a identificação das principais características e limitações deste tipo de informação (Hu et al., 2008):

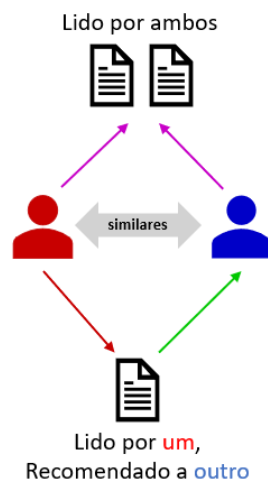
- **Inexistência de *feedback* negativo.** Observando apenas as experiências dos utilizadores, é complicado deduzir, com exatidão e confiança, os produtos que o utilizador não tenha gostado. Por exemplo, se um utilizador não viu um produto, pode indicar que não gostou dele, mas também pode ter sido porque nem o viu.
- **A informação pode ser ruidosa.** As informações recolhidas acerca das experiências dos utilizadores podem, por vezes, não corresponder à sua verdadeira natureza ou motivo, levando o sistema a fazer uma suposição que pode não corresponder à verdade. Por exemplo, o facto de um utilizador ter visto um produto, não significa necessariamente que tenha gostado do produto, pode ter visto e, simplesmente, não gostado.
- **Avaliação à medida.** A avaliação dos sistemas de recomendação que se baseiam em informação implícita necessitam de medidas específicas. Isto significa que, quando é utilizada informação explícita, como os utilizadores avaliam os produtos através de um valor específico, existem métricas claras para medir com exatidão as previsões feitas pelo sistema, como por exemplo, o erro quadrático médio, ou *Mean Square Error*. No entanto, quando são utilizados modelos implícitos, existem outras particularidades que precisam de ser tidas em conta, tais como a disponibilidade do produto, a competitividade entre os produtos, ou, até mesmo informação repetida.
- **Indicação de confiança.** Ao contrário da informação explícita que indica preferência, a informação implícita indica confiança. Nos sistemas baseados em informação implícita os valores numéricos descrevem a frequência de cada experiência dos utilizadores, por exemplo, quantas vezes um determinado produto foi visto, quantas vezes foi comprado, entre outras.

3.2 Técnicas utilizadas em sistemas de recomendação

Tendo em conta as recomendações que cada sistema faz, estes podem ser enquadrados numa das seguintes categorias (Balabanović & Shoham, 1997; Hwangbo et al., 2018; Isinkaye et al., 2015; Khandagale et al., 2016):

- **Filtragem baseada no conteúdo (*Content-Based Filtering*):** As recomendações realizadas baseiam-se apenas nos atributos intrínsecos dos produtos para identificar as recomendações. Por outras palavras, os produtos são recomendados de acordo com a sua similaridade com outros produtos que o utilizador já tenha mostrado alguma preferência;
- **Filtragem Colaborativa (*Collaborative Filtering*):** As recomendações realizadas baseiam-se nas preferências evidenciadas por outros utilizadores similares. Ao invés de medir a similaridade entre os produtos, essa similaridade é medida entre os utilizadores, recomendando produtos sobre os quais tenha mostrado alguma preferência;
- **Filtragem Híbrida:** Técnica que combina ambas as anteriores de forma a ultrapassar as suas limitações.

Filtragem Colaborativa



Filtragem Baseada no Conteúdo



Figura 7 - Ilustração de categorias dos sistemas de recomendação ⁴

⁴ Fonte: <http://datameetsmedia.com/an-overview-of-recommendation-systems>. Acedido em 24-11-2019.

Para sistemas baseados em filtragem colaborativa, é ainda possível a sua subdivisão em dois subtipos. Esta organização encontram-se ilustradas na Figura 8 (Aditya et al., 2016; Khandagale et al., 2016):

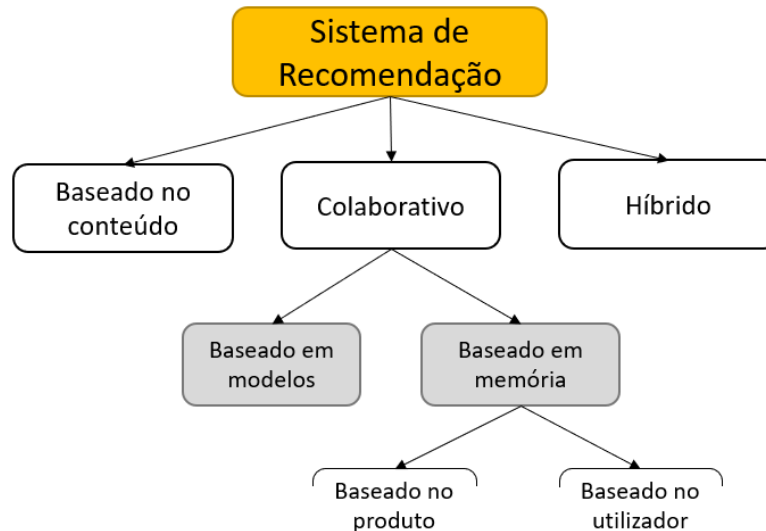


Figura 8 - Esquematização da hierarquia das técnicas

3.2.1 Filtragem baseada no conteúdo

Filtragem baseada no conteúdo utiliza a relação criada entre produtos de forma a identificar aqueles que serão recomendados ao utilizador e são bastante utilizados onde as características dos produtos são bastante discriminadas, tais como: recomendação de páginas web, documentos, notícias, entre outros (Isinkaye et al., 2015).

Para a construção das relações entre os produtos esta técnica pode utilizar vários tipos de modelos, entre os quais (Isinkaye et al., 2015):

- Modelo vetorial em sistemas de recuperação da informação, *Vector Space Model* (VSM) com a utilização do termo inverso da frequência nos documentos, *term frequency-inverse document frequency* (TF-IDF);
- Modelos Probabilísticos, a partir da utilização de *Naïve Bayes*, árvores de decisão, ou redes neuronais.

3.2.1.1 Benefícios na utilização desta técnica

Da utilização desta técnica podem advir várias vantagens, entre as quais (Isinkaye et al., 2015):

- Conservação da privacidade dos utilizadores, dado que apenas é necessária informação relativa ao produto e nenhuma acerca dos dados, ou das características pessoais, dos utilizadores;

- Como não é necessário qualquer dado acerca do utilizador, um produto que seja registado no sistema, desde que tenha os seus atributos com o mínimo de informação, poderá ser imediatamente identificado como uma possível recomendação;
- Se as preferências do utilizador mudarem, esta técnica permite que uma adaptabilidade rápida das recomendações às novas preferências.

3.2.1.2 Limitações na utilização desta técnica

Como a adoção desta técnica não traz só vantagens, foram identificadas as seguintes desvantagens (Isinkaye et al., 2015):

- Sobre especialização (***Overspecialization***). O utilizador apenas irá receber recomendações similares a produtos sobre os quais exerceu algum tipo de atividade e, por isso, caso as atividades do utilizador estagnarem, poderá deixar de existir variedade nas recomendações que lhe são apresentadas;
- Análise de conteúdo limitada (***Limited Content Analysis***). Como esta técnica é muito dependente da informação relativa aos atributos dos produtos, sendo que, é necessária informação descritiva e rica de forma a que a efetividade das recomendações do sistema se mantenha alta.

3.2.2 Filtragem colaborativa

Filtragem colaborativa cria uma lista das melhores recomendações a fazer a um utilizador através da informação relativa aos produtos comprados por utilizadores similares (Balabanović & Shoham, 1997; Hahsler, 2020). Esta informação é armazenada sobre a forma de uma matriz $m \times n$ utilizador-produtos.

Como evidenciado na Figura 8, filtragem colaborativa habitualmente divide-se em dois grupos: baseados em modelo, *model-based*, ou baseados em memória, *memory-based* (Aditya et al., 2016; Adomavicius & Tuzhilin, 2005; Hahsler, 2020; Isinkaye et al., 2015; Khandagale et al., 2016).

3.2.2.1 Filtragem colaborativa baseada em modelo

Sistemas de recomendação baseados em modelos utilizam as experiências dos utilizadores de forma a treinar um modelo que, posteriormente, servirá de molde para gerar as recomendações a apresentar aos utilizadores. Por outras palavras, para a aplicação de técnicas baseadas em modelo, podem ser consideradas vários algoritmos de aprendizagem que através da análise de uma matriz de utilizadores-produtos, origina a identificação de

relações e padrões de forma a originar uma lista com as melhores recomendações (Isinkaye et al., 2015).

3.2.2.1.1 Regras de associação (*Association rules*)

Algoritmos baseados em regras de associação são capazes de identificar a ocorrência de um produto sobre a forma de regras $\{A \Rightarrow B\}$, baseando-se para isso nos produtos presentes na transação. Pode-se tomar como exemplo as seguintes transações de compras: $T1 = \{Alface, Tomate, Cebola\}$, $T2 = \{Alface, Tomate, Alho\}$ e $T3 = \{Alface, Cebola, Tomate, Arroz\}$. Uma possível regra a retirar do conjunto destas transações seria $\{Alface \Rightarrow Tomate\}$, ou seja, um utilizador que compre alface, também compra tomate.

Para a identificação destas regras podem ser utilizadas duas medidas, suporte e confiança (Hahsler, 2020; Isinkaye et al., 2015). Estas medidas servem de filtro para as regras que serão consideradas para a identificação das recomendações a dar aos utilizadores.

- **Suporte** – A frequência com que a sequência existe no conjunto de transações;

$$Sup(X \Rightarrow Y) = \frac{N^{\circ} \text{ de transações que contenham } X \text{ e } Y}{N^{\circ} \text{ total de transações}} \quad (3)$$

- **Confiança** – A frequência com que a regra se verifica ser verdadeira;

$$Conf(X \Rightarrow Y) = \frac{N^{\circ} \text{ de transações que contenham } X \text{ e } Y}{N^{\circ} \text{ de transações que contenham } X} \quad (4)$$

Para o exemplo anterior, a regra indicada apresenta tanto um suporte como uma confiança de 100%, o que indica que 100% das transações contêm alface e tomate (suporte) e 100% dos clientes que compram alface, também compram tomate (confiança). Se fosse considerada a regra $\{Alface, Tomate \Rightarrow Cebola\}$, a análise já seria de que 66,7% das transações contêm alface, tomate e cebola e que 66,7% das transações onde são comprados alface e tomate, também é comprado cebola.

3.2.2.1.2 Análise de agrupamento (*Clustering*)

Algoritmos de *clustering* fragmentam um conjunto de dados de forma a descobrir grupos significativos, também chamados de *clusters*. Após a identificação dos *clusters*, são considerados os dados pertencentes a cada *cluster* para identificar as recomendações a realizar. Estes algoritmos são bastante utilizados em contextos como: reconhecimento de padrões, análise estatística, processamento de imagens, entre outros (Isinkaye et al., 2015; Schafer et al., 1999).

3.2.2.1.3 Árvores de decisão (*Decision Tree*)

Algoritmos baseados em árvores de decisão têm como base a teoria de grafos, onde são identificadas uma série de condições necessárias para que um produto tenha uma determinada classificação. Estes algoritmos treinam sobre um conjunto de produtos já classificados e, após identificação das condições, são aplicados para classificar produtos que ainda não tenham classificação. Com isto, quanto mais rico for o dataset utilizado para treinar, mais preciso serão as classificações dadas (Isinkaye et al., 2015).

3.2.2.2 Sistemas baseados em memória

Por outro lado, sistemas de recomendação baseados em memória acedem diretamente aos dados de forma a realizarem as recomendações aos utilizadores e podem-se dividir em duas categorias: baseados em utilizadores (*User-Based*), ou baseados em produtos (*Item-Based*) (Hahsler, 2020; Hwangbo et al., 2018; Isinkaye et al., 2015).

Sistemas baseados em utilizadores recomendam produtos que outros utilizadores similares preferiram, através do seguinte pressuposto: utilizadores similares entre si têm preferências similaridades. Desta forma, numa primeira fase, ilustrada na Figura 6, são identificados os utilizadores similares ao utilizador para o qual se pretende realizar as recomendações. Ao conjunto desses utilizadores similares dá-se o nome de vizinhança (*neighborhood*). Para a realização das recomendações são então recomendados os produtos que tenham sido comprados pelos utilizadores que compõem a vizinhança e que ainda não tenham sido comprados pelo utilizador para o qual se estão a recomendar os produtos (Hahsler, 2020; Isinkaye et al., 2015).

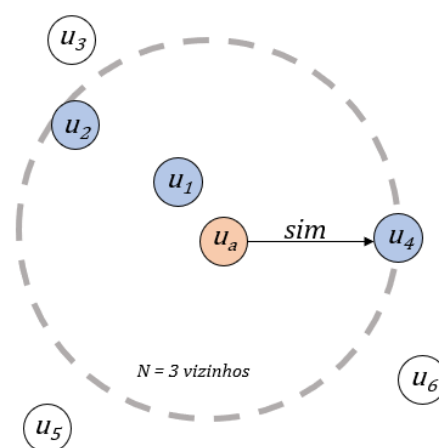


Figura 9 - Representação de uma vizinhança composta por três vizinhos

De forma a identificar os utilizadores que compõem a vizinhança, podem ser utilizadas duas medidas de similaridade (Breese & Kadie, 1998; Hahsler, 2020; Konstan et al., 1997):

- Coeficiente de correlação de *Pearson*, *Pearson correlation coefficient*, dado pela seguinte fórmula:

$$sim_{Pearson}(\vec{x}, \vec{y}) = \frac{\sum_{i \in I} (\vec{x}_i, \vec{x})(\vec{y}_i, \vec{y})}{(|I| - 1) sd(\vec{x}) sd(\vec{y})} \quad (5)$$

- *Cosine similarity*, dado pela seguinte fórmula:

$$sim_{Cosine}(\vec{x}, \vec{y}) = \frac{\vec{x} \cdot \vec{y}}{\|\vec{x}\| \|\vec{y}\|} \quad (6)$$

Sendo sd em (5) o cálculo do desvio padrão e $\|\vec{x}\|$, em (6), a norma L2 do vetor.

Por outro lado, sistemas baseados em produtos recomendam produtos que sejam similares aos que o utilizador preferiu, onde, à semelhança das medidas utilizadas pelos sistemas baseados em modelos, são utilizadas como medidas de similaridade o Coeficiente de correlação de *Pearson* e o *Cosine similarity*, dadas pelas fórmulas (5) e (6), respetivamente (Breese & Kadie, 1998; Hahsler, 2020; Konstan et al., 1997). Karypi (2001) demonstrou que, para sistemas baseados em produtos, a utilização de *cosine similarity* como medida de similaridade, para o cálculo da similaridade entre os produtos, faz com que as recomendações identificadas tenham, em média, maior precisão quando comparadas com as medidas utilizadas nos sistemas baseados em utilizadores.

3.2.2.3 Benefícios na utilização desta técnica

A utilização de uma técnica de filtragem colaborativa pode trazer algumas vantagens, tais como (Perez et al., 2011; Vaidya & Khachane, 2017):

- Dado que não é necessária informação acerca do produto a recomendar, podem ser recomendados, ao utilizador, produtos diferentes daqueles que ele já tenha visto;
- Para sistemas que utilizem esta técnica, não é necessária qualquer informação acerca do produto a recomendar, sendo que as recomendações são obtidas a partir das experiências dos utilizadores;
- Não existe informação inutilizável. Tanto informação explícita, como implícita, podem ser utilizadas.

3.2.2.4 Limitações na utilização desta técnica

Tendo em consideração as especificidades desta técnica, existem várias limitações que advêm da sua utilização, entre elas (Isinkaye et al., 2015; Khandagale et al., 2016; Perez et al., 2011; Vaidya & Khachane, 2017):

- Arranque a frio (**Cold Start**). Quando um item (produto, ou utilizador) é novo para o sistema, como não tem nenhuma informação afeta a ele, quer implícita, quer explícita, não é considerado para recomendação;
- Ovelha negra (**Grey Sheep**). Podem existir utilizadores cujas informações não se assemelham às de mais nenhum outro e, por isso, as recomendações não sejam adequadas. Estes utilizadores designam-se, analogamente, de “ovelhas negras”;
- Disparidade dos dados (**Data Sparsity**). Devido ao número elevado de produtos, pode não existir informação suficiente de forma a que o produto consiga ser recomendado a um utilizador;
- Escalabilidade (**Scalability**). Como a identificação das recomendações cresce de uma forma linear com o número de utilizadores e produtos, a identificação de recomendações pode complicar-se quando estes números são demasiado altos.

3.3 Métodos de avaliação da qualidade de sistemas de recomendação

Nesta seção irão ser analisados três métodos de avaliação da qualidade de um sistema de recomendação: Estudos de Utilizadores, avaliação *offline* e avaliação *online* (Beel et al., 2013; Jeunen, 2019).

3.3.1 Estudos de Utilizadores

Este método de avaliação incide sobre uma revisão quantitativa da perceção da qualidade pelos utilizadores face às recomendações lhes são apresentadas. Por outras palavras, os utilizadores são questionados para avaliarem as recomendações que lhes foram apresentadas e, através dessa avaliação, é verificada a qualidade das recomendações (Beel et al., 2013).

Uma limitação importante deste método a ter em consideração é que, os utilizadores apenas conseguem realizar a sua avaliação face às recomendações que lhes são feitas no momento e, por isso, este método apenas mede a satisfação dos utilizadores e não consegue medir a precisão das recomendações, dado que, os utilizadores que avaliaram, não conseguem ter a perceção se as recomendações que lhes foram apresentadas são efetivamente as que mais se adequam aos seus gostos (Beel et al., 2013).

3.3.2 Avaliação *offline*

Nas avaliações *offline*, a informação é dividida em duas partes: informação para treino e teste. Os modelos aprendem a partir da informação de treino e são avaliados na sua habilidade para recomendar produtos que se encontrem na informação de teste (Jeunen, 2019).

3.3.3 Avaliação online

Neste método as recomendações são apresentadas, em produção, a utilizadores reais, mas não com o objetivo de serem diretamente avaliadas (como acontece no método de estudos de utilizadores), mas sim para que seja observada a aceitação, por parte dos utilizadores, às recomendações. A aceitação de um sistema de recomendação pode ser realizada através da medição de alguns indicadores chave de performance, *Key Performance Indicator* (KPI), tais como: taxa de cliques, *click-through rate* (CTR), transações originadas pelas recomendações, taxa de retenção, entre outros (Jeunen, 2019).

3.3.4 Métricas de avaliação da qualidade do sistema de recomendação

Além dos indicadores apresentados anteriormente, podem ser calculadas outras métricas como forma de avaliação das recomendações dadas, entre elas: erro médio quadrático, *Root Mean Square Error* (RMSE), erro médio absoluto, *Mean Absolute Error* (MAE) e *F-measure*. As mesmas podem classificar-se como métricas estatisticamente precisas, *statistical accuracy metrics*, ou métricas precisas de suporte à decisão, *decision support accuracy metrics* (Isinkaye et al., 2015; B. Sarwar et al., 2001; B. M. Sarwar et al., 1998).

Desta forma, as métricas estatisticamente precisas, nomeadamente RMSE e MAE, comparam as avaliações previstas através da aplicação de diferentes técnicas de filtragem com as avaliações reais dadas pelos utilizadores (Isinkaye et al., 2015).

Por outro lado, as métricas precisas de suporte à decisão, particularmente precisão (*precision*), repetição (*recall*) e *F-measure*, medem de que forma as recomendações ajudam o utilizador a escolher o produto com maior qualidade de entre um leque de produtos disponíveis. Porque estas métricas percebem o processo de recomendações como sendo uma operação binária, o utilizador ou vê o produto recomendado, ou não o vê, avaliando assim as recomendações como sendo boas, ou más (B. M. Sarwar et al., 1998).

Para a medição das métricas e indicadores mencionados, podem-se comparar diferentes algoritmos, ou até mesmo diferentes modelos através da aplicação de testes A/B (Jeunen, 2019). Testes A/B (Gilotte et al., 2018) comparam duas opções dividindo um segmento de utilizadores em dois grupos e fazendo com que um dos grupos utilize uma das opções, enquanto que o outro utilizará a outra, durante um determinado período de tempo. No final de cada experiência, são comparadas as métricas medidas de forma a determinar qual é a solução mais adequada para o negócio.

3.3.4.1 Erro médio quadrático (RMSE)

Mede a diferença entre a avaliação encontrada pelo sistema de recomendação e a avaliação real do produto. Quanto menor for este valor, maior será a precisão das recomendações (Isinkaye et al., 2015; B. M. Sarwar et al., 1998).

$$RMSE = \sqrt{\frac{1}{N} \sum_{u,i} (p_{u,i} - r_{u,i})^2} \quad (7)$$

Sendo N o total de recomendação a considerar, $p_{u,i}$ o valor da avaliação do utilizador u ao item i previsto pelo sistema de recomendação e $r_{u,i}$ o valor real da avaliação feita pelo utilizador u ao item i .

3.3.4.2 Erro médio absoluto (MAE)

Mede o desvio entre as recomendações feitas e os valores especificados pelos utilizadores. Quanto menor for este valor, maior será a precisão das recomendações (B. Sarwar et al., 2001; B. M. Sarwar et al., 1998).

$$MAE = \frac{1}{N} \sum_{u,i} |p_{u,i} - r_{u,i}| \quad (8)$$

Sendo N o total de recomendações a considerar, $p_{u,i}$ o valor da avaliação do utilizador u ao item i previsto pelo sistema de recomendação e $r_{u,i}$ o valor real da avaliação feita pelo utilizador u ao item i .

3.3.4.3 *F-measure*

A métrica *F-measure* é calculada utilizando a precisão (*Precision*) e a sensibilidade, ou revocação (*Recall*), ilustradas pelas equações (9) e (10), respetivamente. Precisão representa a probabilidade de uma recomendação ser relevante para o utilizador. Sensibilidade representa a probabilidade de seleção de uma recomendação, que se pode traduzir na capacidade do sistema em encontrar recomendações que sejam relevantes para os utilizadores (Herlocker et al., 2004; B. M. Sarwar et al., 1998).

$$P = \frac{\text{Produtos corretamente recomendados}}{N^{\circ} \text{ total de produtos recomendados}} \quad (9)$$

$$S = \frac{\text{Produtos corretamente recomendados}}{N^{\circ} \text{ total de recomendações relevantes}} \quad (10)$$

As duas métricas representadas pelas equações (9) e (10), são conflituosas, sendo que, uma alta precisão significa uma baixa sensibilidade e vice-versa. Para encontrar o equilíbrio entre

estas duas métricas, pode utilizar-se uma métrica como *F-measure*, ilustrada na equação (11), tornando a análise a apenas uma métrica que relaciona ambas a precisão com a sensibilidade (Hahsler, 2020; Isinkaye et al., 2015).

$$F - measure = \frac{2 P S}{P + S} \tag{11}$$

Sendo *P* o valor da precisão e *S* o valor da sensibilidade, calculados a partir das equações (9) e (10), respetivamente.

3.3.4.4 Curva ROC

A curva Característica de Operação do Recetor, ou *Receiver Operating Characteristic* (ROC), também pode servir de métrica para o desempenho de sistemas de recomendação. Esta métrica é apresentada sobre a forma de uma curva onde são relacionados o número de recomendações corretas, ou taxa de recomendações positivas verdadeiras (*true positive rate*, - TPR) com o número de recomendações dadas erroneamente, ou taxa das recomendações positivas falsas (*false positive rate* - FPR) (Davis & Goadrich, 2006). Pode-se analisar a eficácia do sistema através do desenho da curva, sendo que, quanto mais aproximada a curva estiver do canto superior esquerdo, mais eficaz é o sistema. Outra análise possível pode ser feita através da área abaixo da curva traçada, onde quanto maior for essa área, maior será a performance do sistema (Davis & Goadrich, 2006; Hahsler, 2020).

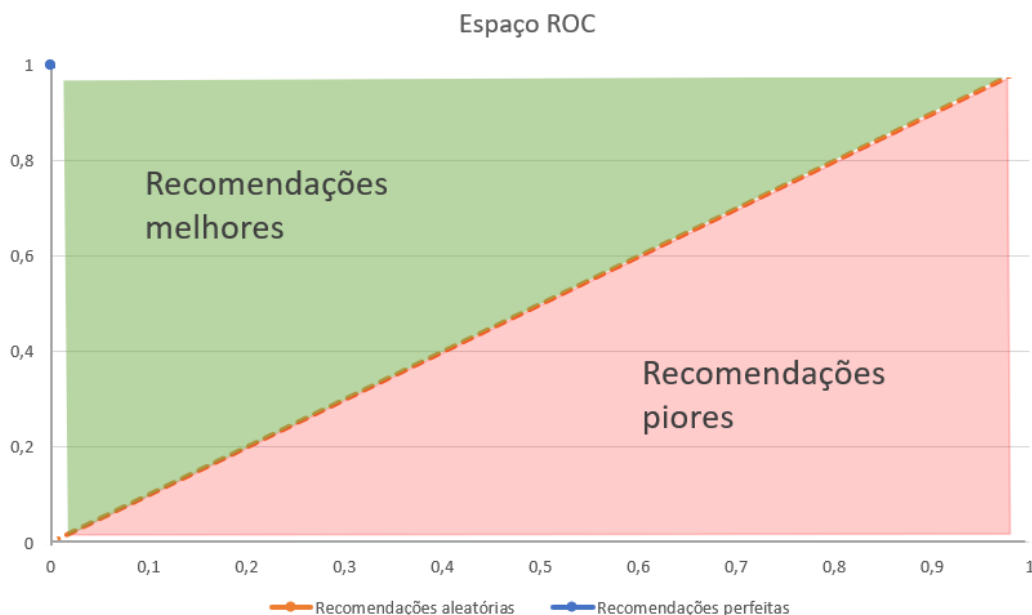


Figura 10 - Representação do espaço ROC e esquematização da análise das curvas

3.4 Aplicações reais de sistemas de recomendação

Podem-se encontrar vários exemplos de aplicações práticas de sistemas de recomendação em utilização em plataformas web. Entre essas plataformas, podem-se ter como exemplos a Amazon.comTM, Netflix e Levi's[®] Strauss.

Para a Amazon.comTM pode-se analisar o exemplo do sistema de recomendação implementado na secção literária. Neste sistema, quando um utilizador se encontra a ver o detalhe de um determinado livro, são-lhe recomendados outros livros que foram comprados por outros utilizadores que compraram o livro que este está a ver no momento, fazendo com que lhe sejam dadas mais opções de escolha sem que, para isso, tenha que navegar para outras páginas (Schafer et al., 1999).

A Netflix utiliza um sistema de recomendação próprio onde se baseia nas interações com o serviço (tais como as visualizações, avaliações que o utilizador fez aos títulos, entre outras), noutros utilizadores com gostos e preferências similares e ainda na informação e características intrínsecas aos títulos, para apresentarem as recomendações aos utilizadores. Além disto, de forma a contornar problemas como o “arranque frio”, abordado nos capítulos posteriores, a Netflix implementou a estratégia de, sempre que um novo utilizador se regista na plataforma, é-lhe pedido que indique alguns títulos que sejam do seu agrado, para que, quando entre pela primeira vez na plataforma, o sistema já tenha informação suficiente para lhe recomendar, de uma forma precisa, títulos que possa ver (*Funcionamento do sistema de recomendações da Netflix*, 2019).

A Levi's[®] Strauss disponibiliza uma página onde o utilizador é encaminhado por um questionário onde seleciona o seu género e os looks que mais se adequam aos seus gostos. No final desta seleção, é-lhe apresentado um quadro com várias inspirações baseando-se nas suas escolhas anteriores - <https://www.styledbylevis.levi.com>.

3.4.1 Trabalhos relacionados

Diversas pesquisas foram realizadas nesta área, nas quais foram propostos vários sistemas com relevância para o problema a ultrapassar neste projeto, ou adotando uma abordagem distinta da tradicional.

Shen, Lieberman, & Lam, (2007) investigaram um sistema de recomendação orientado a situações, ou cenários específicos. Por outras palavras, o sistema recomenda produtos cujas características se enquadram nos objetivos que o utilizador tem para uma determinada situação.

Hwangbo et al., (2018) propuseram a implementação de um sistema de recomendação de produtos de retalho baseado não só na filtragem colaborativa, mas também em algumas características dos produtos, nas compras realizadas em lojas físicas e nos dados de cliques das lojas online. Outro facto que o sistema tem em consideração é a decadência, ao longo do tempo, das preferências dos utilizadores. Para isso, foi também proposta uma função que diminui a intensidade dada às preferências à medida que o tempo vai passando.

Karypi, (2001) apresentou um algoritmo de recomendações *model-based* que se baseia em similaridades entre produtos principalmente mais rápidos e, por isso, capazes de recomendar os produtos em tempo real e sem limitações à quantidade de informação necessária.

Khandagale et al., (2016) implementaram um sistema de recomendação baseadas em padrões encontrados no histórico de transações dos utilizadores através de técnicas de análise de padrões sequenciais. Análise de padrões sequenciais visa a identificar de padrões sobre a forma de subsequências numa base de dados.

4 Propostas de soluções

Nesta secção serão analisados os requisitos para a solução final e, a partir dessa análise, serão apresentadas duas propostas de arquitetura para a solução final. Ambas as soluções apresentam as suas vantagens e desvantagens, pelo que serão descritos os seus componentes, bem como aquilo que as diferencia. Por último, serão mencionadas as tecnologias e abordagens possíveis a considerar para alguns componentes das soluções apresentadas.

4.1 Análise de requisitos

A plataforma de E-commerce faz registo das experiências dos utilizadores e pretende-se que estas sejam comunicadas individualmente à medida que forem sendo despoletadas na plataforma, ou em grupo através de uma determinada frequência. Estas experiências passam por interações, ou eventos entre os utilizadores e os produtos. Cada evento tem o seu peso, sendo que uns têm uma importância maior para as recomendações que outros. Estes eventos, do mais importante para o menos, são os seguintes:

1. Utilizador X comprou o produto Y;
2. Utilizador X adicionou o produto Y ao seu carrinho;
3. Utilizador X clicou no produto Y;
4. Utilizador X viu o produto Y.

Com estas experiências pretende-se assim o desenvolvimento de um sistema de recomendação de produtos. Numa fase inicial, o sistema apenas irá utilizar os eventos de compra dos utilizadores, ou seja, o seu histórico de compras. No entanto, pretende-se que seja capaz de, no futuro, através da implementação de um novo mecanismo de recomendação, ou de alterações ao mecanismo existente, fazer uso dos restantes eventos. Desta forma, a solução terá de assegurar as seguintes funcionalidades, por ordem de necessidade:

1. Dado um utilizador, recomendar quais os produtos mais prováveis deste comprar;
2. Dado um utilizador, enquadrar o mesmo num segmento;
3. Para cada segmento, criar uma lista pronta a exportar com todos os produtos ordenados pela probabilidade de serem comprados (listas atualizadas uma vez por dia).

| Número do requisito | Descrição |
|---------------------|---|
| REQF001 | Integrar as experiências dos utilizadores |
| REQF002 | Identificar o cliente num segmento |
| REQF003 | Identificar os produtos a recomendar por utilizador |
| REQF004 | Identificar os produtos a recomendar por grupo |
| REQF005 | Exportar as recomendações por grupo |
| REQNF001 | Comunicações entre a plataforma de E-commerce e a solução através de pedidos HTTP |

Tabela 8 - Descrição dos requisitos funcionais e não funcionais

4.2 Apresentação das soluções

Serão apresentados diagramas de implantação de duas propostas para a solução ao problema apresentado. Em todos os diagramas, encontram-se evidenciados dois componentes comuns: *BaseDeDados* e *SistemaRecomendacoes*.

A *PlataformaEcommerce* é a plataforma já existente que contém toda a lógica de negócio e que se irá servir das funcionalidades disponibilizadas pela solução.

O componente *SistemaRecomendacoes* contém toda a lógica necessária para a identificação e comunicação das recomendações a fazer aos utilizadores.

O componente *BaseDeDados* será uma base de dados responsável pela persistência de dados de configuração necessários para o sistema de recomendação, bem como quaisquer outros dados necessários. Para esta base de dados podem-se optar por duas tecnologias distintas: MongoDB®, ou MySQL™ (sistema de gestão de bases de dados utilizado pela Redicom).

4.2.1 Primeira Solução

A primeira solução é a que apresenta menos componentes, pelo que apenas contém dois componentes. Um dos componentes é uma base de dados responsável pela persistência dos dados e comunicação das recomendações com a plataforma de E-commerce. O outro componente, *SistemaRecomendacoes*, é responsável por identificar todas as recomendações de cada utilizador registado na base de dados e o seu respetivo histórico de compras. Nesta solução as recomendações são geradas pelo *SistemaRecomendacoes* todos os dias e

persistidas na base de dados para, posteriormente, serem consultadas pela plataforma de E-commerce.

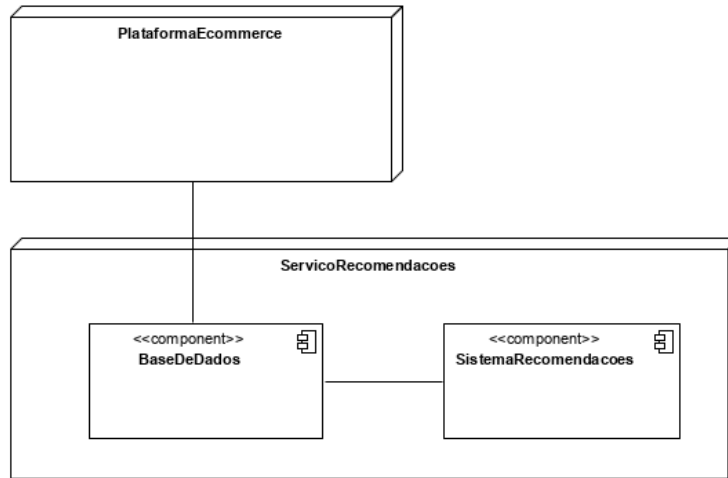


Figura 11 - Diagrama de implantação da primeira solução

4.2.2 Segunda Solução

Para a segunda solução os componentes são os mesmos, mas com uma alteração ao componente *SistemaRecomendacoes*. Este componente passa a disponibilizar um conjunto de interfaces capazes de comunicar com a plataforma de E-commerce de forma a desacoplar a implementação da solução da plataforma de E-commerce. Desta forma, o componente *SistemaRecomendacoes* fica responsável por conter todas as regras para a identificação das recomendações, bem como todas as interfaces para comunicação com a plataforma de E-commerce. A base de dados continua a existir, mas desta forma, apenas para persistir os dados necessários para a identificação das recomendações.

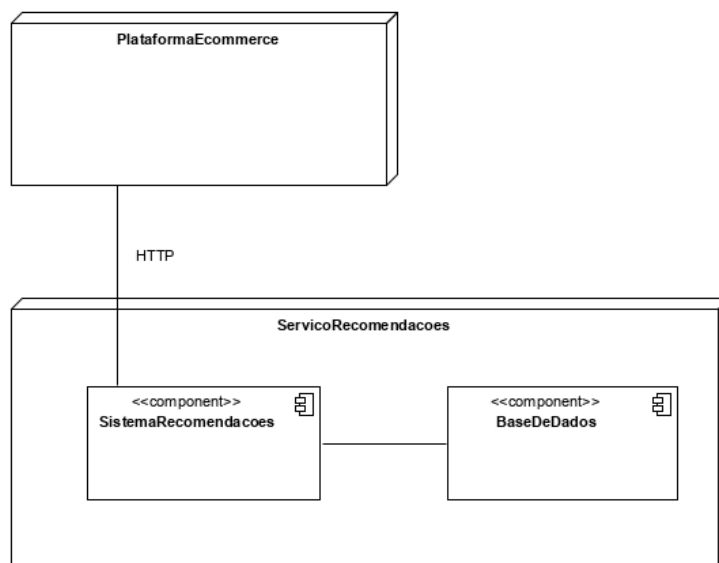


Figura 12 - Diagrama de implantação da segunda solução

4.2.3 Terceira solução

Para a terceira solução a base é a mesma que a da segunda, mas com o objetivo de diminuir ainda mais a dependência entre os sistemas, mas com o senão de poder aumentar à complexidade da solução.

Para isso a comunicação das experiências dos utilizadores não será feita de uma forma direta para o sistema de recomendação, mas sim através do módulo *FilaEventos*, que nada mais é do que um intermediário de mensagens (*Message Broker*). Caso exista a necessidade de reaproveitamento desta informação para outros sistemas, não haja a necessidade de implementação na plataforma de E-commerce já existente, basta que o novo sistema consuma os eventos que já estão a ser comunicados.

De forma a separar melhor as responsabilidades afetas a cada componente, nesta solução o componente *SistemaRecomendacoes* apenas disponibilizará à plataforma de E-commerce as interfaces para obtenção de recomendações.

O componente *IntegradorEventos* será responsável por consumir os eventos que vão sendo comunicados para a *FilaEventos* e persistir os mesmos na base de dados. Por fim, o componente *SistemaRecomendacoes* será o componente que contem todas as regras necessárias para identificar as recomendações a comunicar à plataforma de E-commerce.

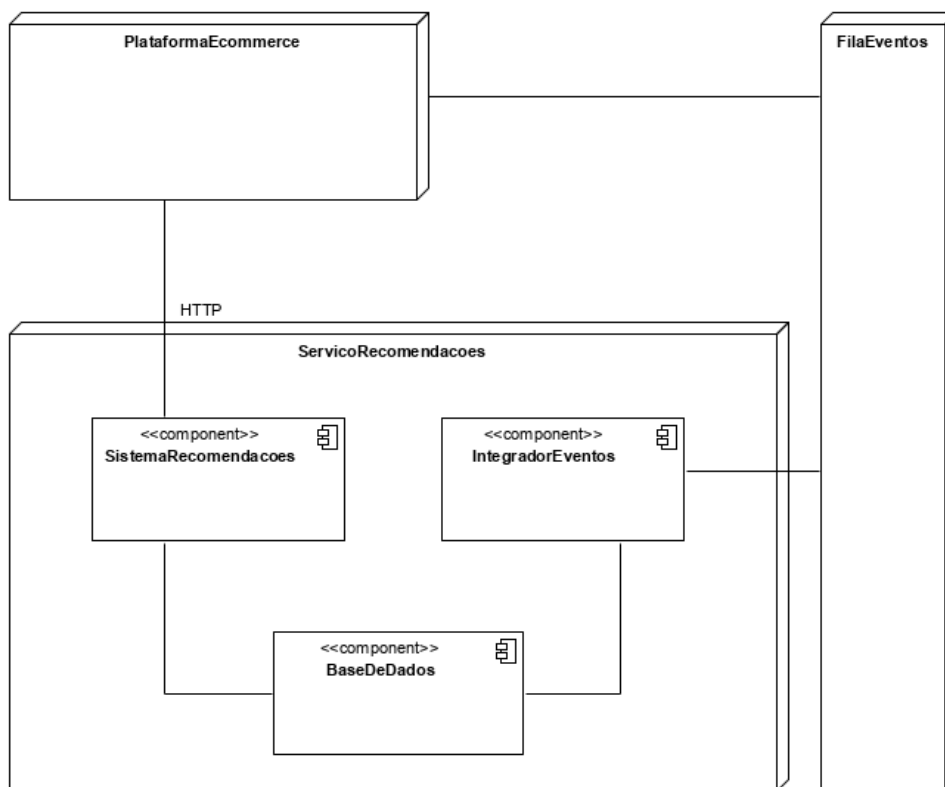


Figura 13 - Diagrama de implantação da terceira solução

4.2.3.1 Intermediário de mensagens (*Message Broker*)

Para a implementação deste intermediário, foram considerados dois sistemas de código aberto populares no paradigma de publicador e subscritor: Apache Kafka® e RabbitMQ. Este paradigma permite o desacoplamento entre os sistemas na medida em que as comunicações são feitas através de mensagens categorizadas, onde uma entidade publicadora envia uma qualquer mensagem com uma categoria e um subscritor subscreve a essa categoria, independentemente do teor da mensagem (Dobbelaere & Esmaili, 2017).

As principais diferenças entre os sistemas encontram-se na performance e complexidade de configuração dos mesmos, sendo que a performance do Apache Kafka® é melhor que do RabbitMQ, mas, em contrapartida, é um sistema mais complexo de configurar (Dobbelaere & Esmaili, 2017).

5 Casos de estudo

Nos capítulos anteriores verificou-se a existência de vários algoritmos com determinadas vantagens e desvantagens entre eles. De entre todos os algoritmos existentes, é necessário verificar qual aquele que apresenta melhores resultados e, por consequência, consegue recomendar aos utilizadores os produtos mais prováveis de serem comprados. Para isso foi feita uma análise ao histórico de compras dos utilizadores de forma a caracterizar o comportamento das suas compras e, de seguida, foram realizadas várias avaliações aos produtos recomendados utilizando diferentes métodos em diferentes amostras. O negócio do cliente analisado é direcionado para a venda de peças de vestuário.

5.1 Análise dos utilizadores

Devido ao elevado número de compras realizadas na plataforma, foi necessário reduzir os dados a considerar para a caracterização dos utilizadores. Para manter uma variedade de dados, com relativa abundância, serão utilizadas as compras relativas a , aproximadamente, cem utilizadores com mais compras na plataforma. Assim, a amostra é composta por um total de 40.393 produtos comprados em 23.574 encomendas.

Para a composição da amostra acima mencionada, foram recolhidos os seguintes dados:

- Identificador da encomenda;
- Identificador do utilizador que fez a encomenda;
- País no qual foi efetuado a encomenda;
- Data da encomenda;
- Identificador da compra;
- Identificador do produto;
- Referência do produto;
- Cor do produto;
- Género do produto;
- Família do produto;
- Valor do produto.

A partir da análise da amostra indicada verificou-se que, em média, cada utilizador, por encomenda, compra cerca de 2 produtos (valor real de 1,71) a cada 11 dias, gastando, aproximadamente, 63.16€.

Através da análise do Gráfico 1 pode verificar-se que existe uma tendência para a compra de algumas peças específicas, nomeadamente calças e t-shirts. Todas as restantes peças apresentam um número de compras semelhantes. Isto pode indicar uma tendência para que os artigos mais populares sejam calças e t-shirts.

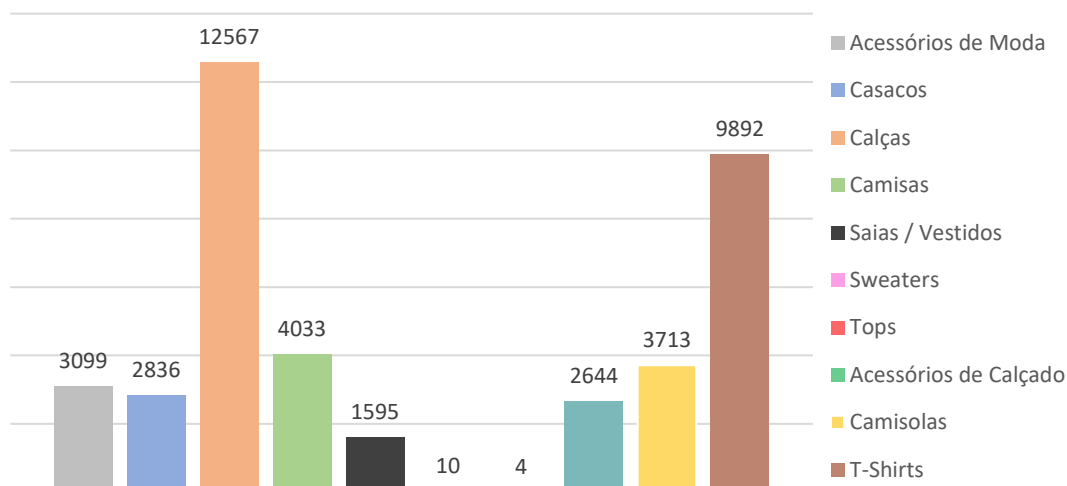


Gráfico 1 - Número de produtos comprados em cada família

Pode-se ainda verificar, através da análise do Gráfico 2 que existe uma tendência para a compra de artigos de mulher. Em conjunto com os dados do Gráfico 1 é possível traçar um perfil das peças mais vendidas na plataforma, ou seja, calças e t-shirts de mulher.

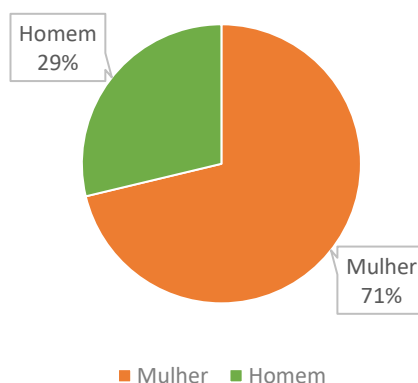


Gráfico 2 - Percentagem de produtos comprados em cada género

No Gráfico 3 é visível que, além da tendência de compra ser de produtos de mulher, os valores gastos, em média, nos produtos deste género são também ligeiramente superiores aos que são gastos em produtos de homem.

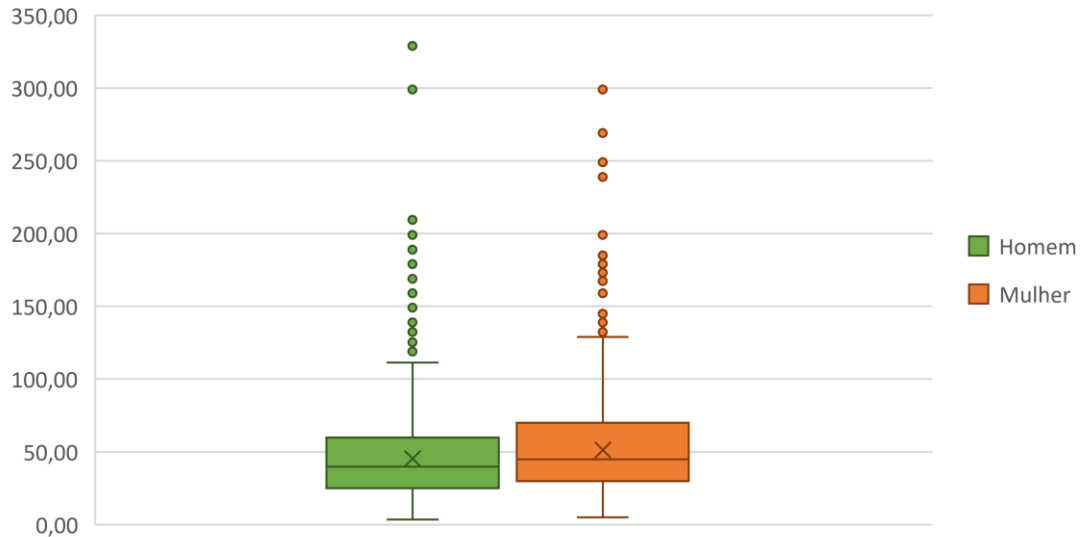


Gráfico 3 - Valores gastos em produtos para cada género

Por último, pode verificar-se que, através da informação disponibilizada no Gráfico 4, não existe nenhuma relação visível entre o valor médio gasto e o número médio de dias até à próxima compra de cada cliente.

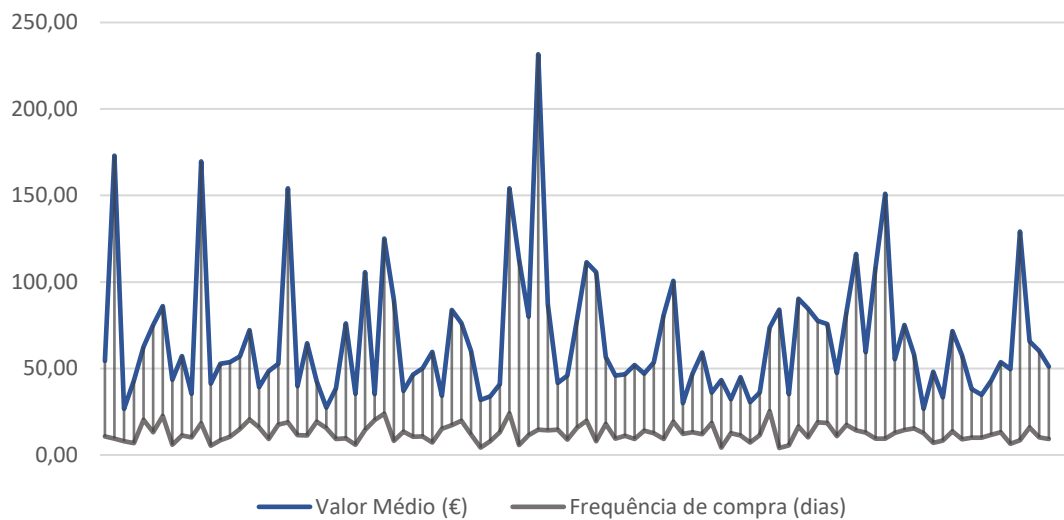


Gráfico 4 - Relação entre o valor médio gasto por encomenda e o número de dias até à próxima compra

Além de toda a análise descrita acima, foi também possível aplicar o método baseado em regras associativas para verificar que padrões de compra eram possíveis identificar nos utilizadores. À semelhança das análises anteriores, a aplicação do método teve como objetivo a identificação de alguns padrões de compras relativas a famílias dos produtos. Dito isto e,

através da aplicação do algoritmo APRIORI com um suporte igual de 4% e um grau de confiança igual a 80%, foi possível identificar as seguintes regras:

- $\{Calças, Camisas \Rightarrow Acessórios de Moda\}$ - sup=7%, conf=100%
- $\{Acessórios de Moda, Camisas \Rightarrow Calças\}$ - sup=7% ,conf =87,5%
- $\{Calças, Acessórios de Calçado \Rightarrow Acessórios de Moda\}$ - sup=10%, conf=91%
- $\{Camisas, Acessórios de Calçado \Rightarrow Acessórios de Moda\}$ - sup=6%, conf=100%
- $\{Calças, Camisolas \Rightarrow Acessórios de Moda\}$ - sup=5%, conf=100%
- $\{Acessórios de Moda, Camisolas \Rightarrow Calças\}$ - sup=5%, conf=100%
- $\{Camisas, Acessórios de Calçado \Rightarrow Calças\}$ - sup=5%, conf=84%
- $\{Calças, Camisas, Acessórios de Calçado \Rightarrow Acessórios de Moda\}$ - sup=5%, conf=100%
- $\{Acessórios de Moda, Camisas, Acessórios de Calçado \Rightarrow Calças\}$ - sup=5%, conf=84%
- $\{Camisas, Acessórios de Calçado \Rightarrow Acessórios de Moda, Calças\}$ - sup=5%, conf=84%

Através das regras enumeradas é possível verificar que as que apresentam um maior suporte e grau de confiança são aquelas que apresentam a família de produtos mais comprada, ou seja, calças. Além disto, pode também verificar-se uma potencial oportunidade para o negócio dado que as regras que apresentam um grau de confiança maior são as que preveem a compra de um acessório de moda. Através desta análise pode verificar-se que, estes produtos que não se destacam quanto ao número de compras, têm o potencial de, através da sua recomendação, aumentar as suas vendas, aumentando, assim, as vendas para o cliente.

5.2 Avaliação dos métodos de recomendação

Para a avaliação dos métodos de recomendação foram utilizadas três amostras com diferentes características. Essas amostras foram utilizadas na avaliação de cinco métodos de recomendação disponibilizados pela biblioteca *recommenderlab*, sendo estes:

- Regras associativas (AR);
- Produtos aleatórios (RI);
- Produtos mais populares (PI);
- Filtragem colaborativa baseada no produto (IBCF);
- Filtragem colaborativa baseada no utilizador (UBCF).

Para os métodos AR, IBCF e UBCF foi feita, primeiramente, uma análise de sensibilidade aos parâmetros, utilizando as configurações presentes na Figura 24, de forma a que seja encontrada a configuração que melhores recomendações realizou. Com a análise de qual configuração obteve melhores resultados, todos os métodos foram avaliados entre si de forma verificar qual o método que realiza melhores recomendações e, por consequência, entrará em produção. Todo este processo, esquematizado na Figura 14, foi aplicado para cada uma das três amostras.

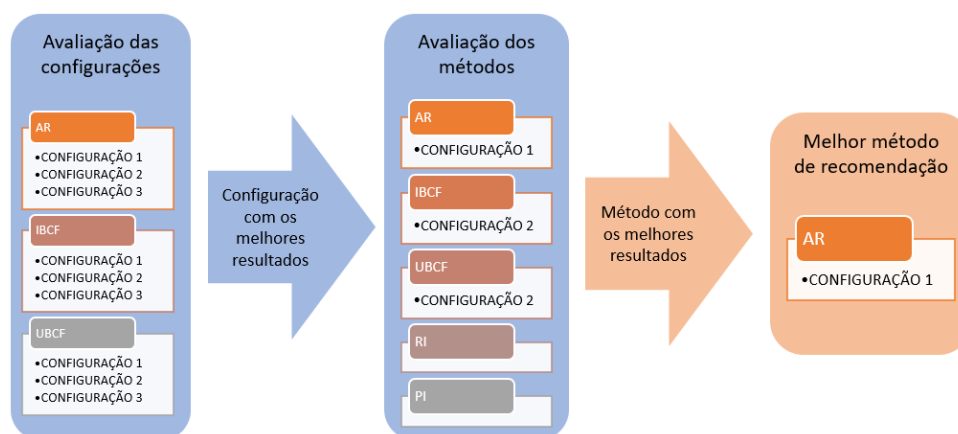


Figura 14 - Processo de análise para selecionar o melhor método de recomendação

Tendo identificado o método que, para cada amostra, melhores resultados apresentou, será considerada a análise feita aos utilizadores no capítulo 5.1 para selecionar o método e respetiva configuração, se for o caso, que entrará em produção.

5.2.1 Identificação das amostras a utilizar

Todas as avaliações foram feitas sobre três amostras, caracterizadas na Tabela 9, cada uma com as suas particulares. Foram selecionadas estas amostra para que os métodos sejam avaliados sobre contextos diferentes e para que seja possível a identificação de determinados comportamentos que cada método possa ter mediante as características de cada amostra.

| | Nº de Linhas (Utilizadores) | Nº de colunas (Produtos) | Nº de compras | Média de compras por utilizador | Média de compras por produto |
|------------------|--------------------------------|-----------------------------|------------------|---------------------------------------|------------------------------------|
| Amostra 1 | 87 | 48 | 694 | 8 | 14 |
| Amostra 2 | 101 | 944 | 1450 | 14 | 2 |
| Amostra 3 | 676 | 98 | 1885 | 3 | 19 |

Tabela 9 - Resumo das características de cada amostra

Como indicado, cada amostra tem as suas características próprias, mas existem alguma regras que têm de ser cumpridas de forma a que a matriz esteja em conformidade para ser utilizada nos respetivos métodos de recomendação. Todas as matrizes utilizadas são binárias, pelo que apenas contêm os valores um (*TRUE*), ou zero (*FALSE*), caso o utilizador tenha, ou não, comprado o produto, respetivamente. Dado que a matriz apenas contém estes dois valores, não é possível a representação do cenário onde o utilizador compre mais que uma vez o mesmo produto. Com isto, é necessário realizar uma filtragem aos dados de forma a remover, para cada utilizador, todas as compras duplicadas. Com este pré-processamento a matriz, representada na Figura 15, fica em conformidade para que seja utilizada para treinar o modelo do respetivo método de recomendação.

| | 100256 | 100301 | 100347 | 100353 | 107507 | 109036 | 109038 | 109057 | 109058 | 109071 | 109411 | 109462 | 109492 | 110633 | ID do Produto |
|----|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|---------------|
| 1 | TRUE | FALSE | FALSE | FALSE | TRUE | FALSE | TRUE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | |
| 2 | FALSE | FALSE | FALSE | FALSE | FALSE | TRUE | FALSE | FALSE | FALSE | FALSE | FALSE | TRUE | FALSE | TRUE | |
| 3 | TRUE | TRUE | TRUE | FALSE | FALSE | TRUE | TRUE | FALSE | FALSE | FALSE | FALSE | TRUE | FALSE | FALSE | |
| 4 | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | TRUE | FALSE | FALSE | FALSE | FALSE | TRUE | FALSE | FALSE | |
| 5 | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | TRUE | FALSE | TRUE | |
| 6 | TRUE | FALSE | FALSE | FALSE | FALSE | TRUE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | |
| 7 | TRUE | FALSE | FALSE | FALSE | TRUE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | TRUE | FALSE | FALSE | |
| 8 | FALSE | FALSE | FALSE | FALSE | TRUE | FALSE | TRUE | TRUE | FALSE | FALSE | FALSE | FALSE | FALSE | TRUE | |
| 9 | TRUE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | TRUE | FALSE | FALSE | |
| 10 | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | TRUE | |
| 11 | TRUE | FALSE | FALSE | FALSE | TRUE | FALSE | TRUE | FALSE | FALSE | FALSE | FALSE | FALSE | TRUE | FALSE | |
| 12 | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | TRUE | FALSE | FALSE | FALSE | FALSE | TRUE | FALSE | |
| 13 | TRUE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | |
| 14 | TRUE | FALSE | FALSE | FALSE | TRUE | TRUE | FALSE | FALSE | TRUE | FALSE | FALSE | FALSE | FALSE | FALSE | |
| 15 | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | TRUE | FALSE | |
| 16 | FALSE | FALSE | FALSE | FALSE | FALSE | TRUE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | |
| 17 | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | TRUE | FALSE | FALSE | |
| 18 | FALSE | FALSE | FALSE | FALSE | TRUE | FALSE | TRUE | FALSE | FALSE | FALSE | FALSE | TRUE | FALSE | TRUE | |
| 19 | FALSE | FALSE | FALSE | FALSE | TRUE | FALSE | TRUE | FALSE | FALSE | FALSE | FALSE | TRUE | FALSE | TRUE | |
| 20 | FALSE | TRUE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | FALSE | TRUE | FALSE | TRUE | |

Figura 15 - Excerto de uma matriz utilizada para treinar o sistema de recomendação

Com tudo isto, para a primeira amostra foi então utilizada como base, a informação relativa ao histórico de compras dos cem utilizadores com mais compras na plataforma. Embora o objetivo desta amostra seja a existência de um grande número de compras, para as avaliações a realizar, as mesmas estão numa quantidade excessiva. Assim, as compras da amostra foram reduzidas para os cinquenta produtos mais comprados, de forma a que as avaliações sejam feitas num período apropriado.

Desses dados, apenas ficaram os utilizadores com mais que uma compra, perfazendo assim a matriz, representada na Figura 16, composta por oitenta e sete utilizadores, em cada linha, e quarenta e oito produtos, por cada coluna.

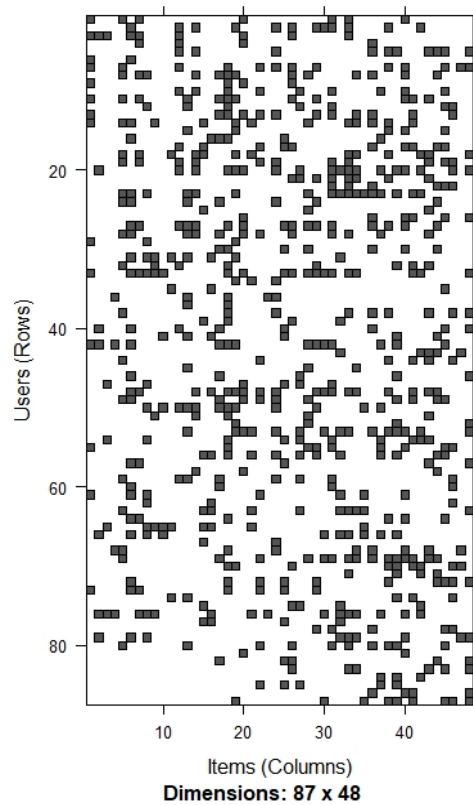


Figura 16 - Matriz dos dados da primeira amostra

Para a segunda amostra foi utilizado todo o histórico de compras de, aproximadamente, cem utilizadores obtido durante os meses de julho e agosto de 2019. Removendo dos dados os utilizadores que apenas têm uma compra, a matriz da segunda amostra, representada na Figura 17, é composta por cento e um utilizadores, por linha e novecentos e quarenta e quatro produtos, por cada coluna.

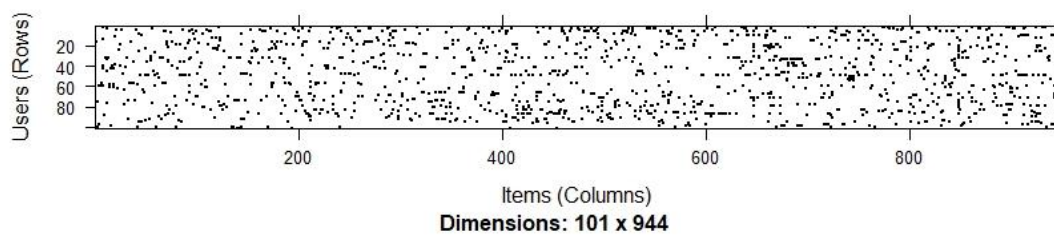


Figura 17 - Matriz dos dados da segunda amostra

A terceira e última amostra tem por base o histórico de compras relativo aos cem produtos mais comprados, aproximadamente, durante os meses de julho e agosto de 2019. Desses dados, a amostra é constituída pelos clientes que têm mais que duas compras feitas. Com estes dados, são, novamente, removidos os utilizadores que apenas têm uma compra,

fazendo com que a matriz, ilustrada na Figura 18, seja composta por seiscentos e setenta e seis utilizadores por linha e noventa e oito produtos, um por coluna.

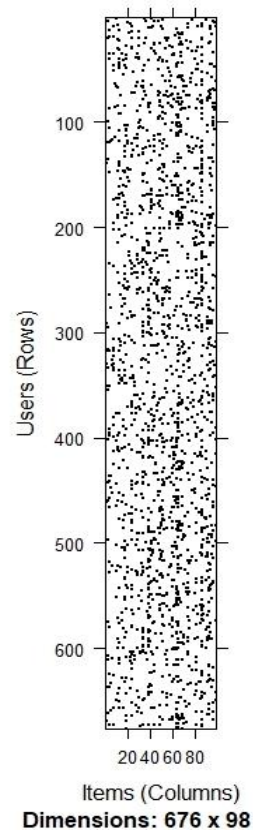


Figura 18 - Matriz dos dados da terceira amostra

O objetivo da utilização destas amostras passa pela avaliação dos algoritmos utilizando diferentes contextos para o efeito. A primeira amostra tenta simular um cenário ideal, pelo que contém as compras dos produtos mais populares da plataforma, em conjunto com as sequências de compras dos utilizadores que mais compras fazem. As restantes amostras tentam combinar um cenário onde predominam produtos e utilizadores, respetivamente, sendo que cada uma delas contém o que era característico da primeira amostra, ou seja, a segunda amostra contém os produtos mais populares e a terceira as sequências de compras dos utilizadores, sem que as duas, deliberadamente, sejam coexistentes.

5.2.2 Métodos de recomendação

Com o objetivo de encontrar o melhor método de recomendação para utilizar em produção serão avaliados os cinco métodos seguintes:

- Regras associativas (AR);
- Produtos aleatórios (RI);

- Produtos mais populares (PI);
- Filtragem colaborativa baseada no produto (IBCF);
- Filtragem colaborativa baseada no utilizador (UBCF).

```

43
44 # Set up List of Algorithms
45 algorithms <- list(
46   "association rules" = list(name = "AR", param = list(supp = 0.02, conf = 0.4)),
47   "random items"      = list(name = "RANDOM", param = NULL),
48   "popular items"    = list(name = "POPULAR", param = NULL),
49   "item-based CF"    = list(name = "IBCF", param = list(k = 20)),
50   "user-based CF"    = list(name = "UBCF", param = list(nn = 20))
51 )
52

```

Figura 19 - Exemplo de uma lista de métodos a utilizar para recomendação

Como detalhado no Capítulo 3.2.2.1.1, um sistema de recomendação baseado em regras associativas utiliza o histórico dos utilizadores para encontrar padrões de compras e, a partir desses padrões, recomendar produtos aos utilizadores. De forma a encontrar a melhor configuração para o suporte e grau de confiança a utilizar para o método, foi realizada uma análise de sensibilidade aos parâmetros.

O método de recomendação de produtos aleatórios, como o próprio nome indica, é um método que escolhe, aleatoriamente, os produtos a recomendar ao utilizador. Este método apenas tem em consideração o histórico de compras do utilizador para que não sejam recomendados produtos já comprados.

Recomendação de produtos mais populares recomenda os produtos mais comprados, tendo em consideração o histórico de compras dos utilizadores para que não sejam recomendados produtos que o utilizador já tenha comprado no passado.

Filtragem colaborativa baseada no produto e baseada no utilizador podem ser configuradas para utilizar N vizinhos, e, como medidas de similaridade, podem utilizar o coeficiente de correlação de *Pearson*, ou o *Cosine similarity*.

Filtragem colaborativa baseada no utilizador utiliza a informação das compras dos utilizadores de forma a verificar outros N utilizadores com compras similares às deles. A partir das compras desses N utilizadores são realizadas as recomendações, ou seja, são recomendados os produtos comprados pelos N utilizadores que ainda não tenham sido comprado pelo utilizador, como ilustrado na Figura 20.

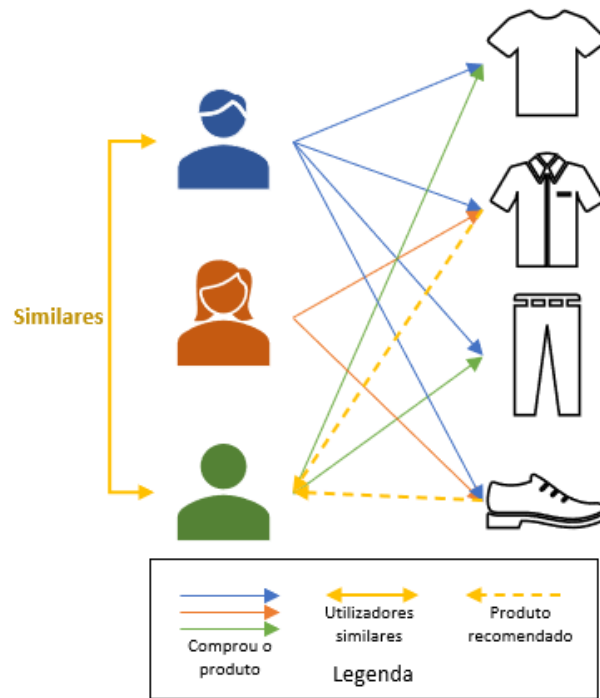


Figura 20 - Esquema do método de filtragem colaborativa baseada em utilizadores

Filtragem colaborativa baseada nos produtos, esquematizada na Figura 21, utiliza a informação dos produtos de forma a verificar outros N produtos que sejam similares entre si. Para realizar as recomendações, são identificados os produtos comprados pelo utilizador e são recomendados aqueles que compõem as suas vizinhanças, ou seja, produtos similares aos que o utilizador comprou.

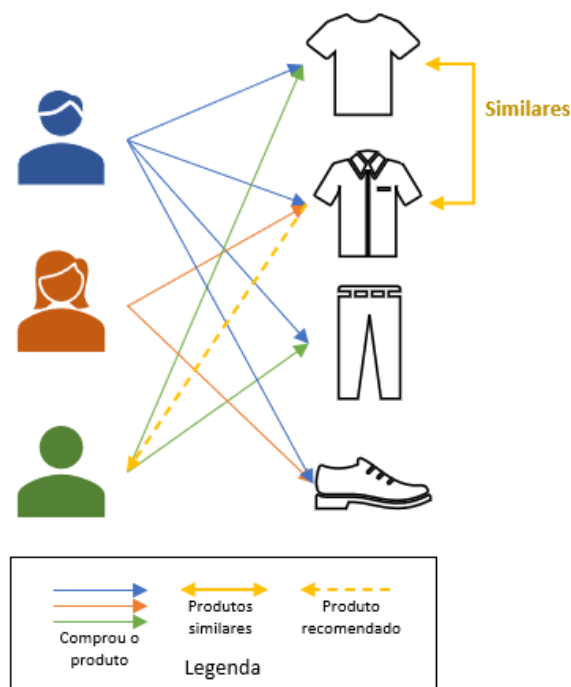


Figura 21 - Esquema do método de filtragem colaborativa baseada em produtos

5.2.3 Processo de avaliação dos métodos de recomendação

Para todas as avaliações efetuadas foi utilizado um esquema de avaliação cruzada (*k-fold cross-validation*) em conjunto com um protocolo de todos menos um (“*all but one*”) produto. A avaliação incidiu sobre seis listas com uma, três, cinco, dez, quinze e vinte recomendações realizadas.

O esquema de avaliação cruzada, ilustrado na Figura 22, divide os utilizadores, aleatoriamente, em *k* partes (*folds*), sendo que cada parte contém aproximadamente o mesmo número de utilizadores. São realizadas *k* avaliações onde, em cada uma, é utilizada uma parte para teste e as restantes partes para treino. Com isto, a abordagem garante que todos os utilizadores são utilizados, pelo menos uma vez, para teste, fazendo com que os resultados sejam mais robustos (Hahsler, 2020).

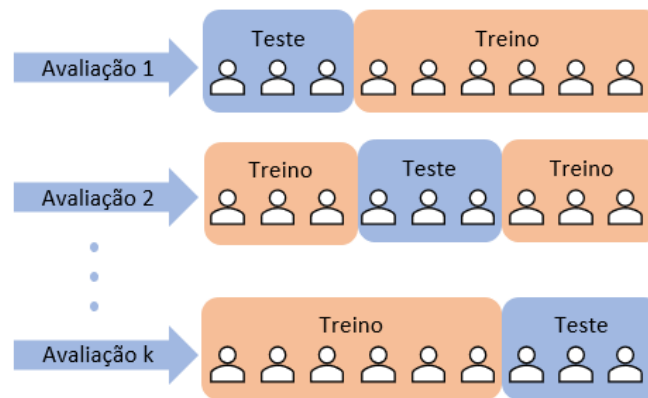


Figura 22 - Esquema de avaliação cruzada

Foi também adotado o protocolo de todos menos um (“*all but one*”) produto (Hahsler, 2020), que por sua vez faz com que, para os utilizadores que pertençam às partes utilizadas para treino, sejam consideradas todas as compras à exceção de uma (escolhidas aleatoriamente).

Para complementar o esquema anterior foi utilizado um *k* igual a cinco. Este valor faz com que exista um equilíbrio entre o número de utilizadores considerados para teste e o número de avaliações que são necessárias fazer para que todos os utilizadores sejam testados. Encontra-se visível na Figura 23, o código utilizado para o esquema descrito.

```

52
53 # Creates an evaluationScheme object from a data set
54 evaluation_scheme <- evaluationScheme(
55     ratings_matrix,
56     method="cross-validation", # Character string defining the evaluation method to use
57     k=5, # Number of folds/times to run the evaluation (defaults to 10 for cross-validation)
58     given=-1 # Single number of items given for evaluation
59 )
60

```

Figura 23 - Configuração utilizada para as avaliações

Numa primeira fase foi feita uma análise de sensibilidade aos parâmetros para os seguintes métodos de recomendação: regras associativas, filtragem colaborativa baseada nos produtos e baseada nos utilizadores. As configurações avaliadas podem ser visíveis na Figura 24.

```

46 algorithms_ar <- list(
47   "AR 1 20" = list(name = "AR", param = list(supp = 0.01, conf = 0.2)),
48   "AR 1 50" = list(name = "AR", param = list(supp = 0.01, conf = 0.5)),
49   "AR 1 90" = list(name = "AR", param = list(supp = 0.01, conf = 0.9)),
50   "AR 2 20" = list(name = "AR", param = list(supp = 0.02, conf = 0.2)),
51   "AR 2 50" = list(name = "AR", param = list(supp = 0.02, conf = 0.5)),
52   "AR 2 90" = list(name = "AR", param = list(supp = 0.02, conf = 0.9))
53 )
54
55
56 ## Set up List of Algorithms
57 algorithms_ib <- list(
58   "IBCF P 10" = list(name = "IBCF", param = list(k = 10, method="pearson")),
59   "IBCF C 10" = list(name = "IBCF", param = list(k = 10, method="cosine")),
60   "IBCF P 30" = list(name = "IBCF", param = list(k = 30, method="pearson")),
61   "IBCF C 30" = list(name = "IBCF", param = list(k = 30, method="cosine")),
62   "IBCF P 50" = list(name = "IBCF", param = list(k = 50, method="pearson")),
63   "IBCF C 50" = list(name = "IBCF", param = list(k = 50, method="cosine"))
64 )
65
66 # Set up List of Algorithms
67 algorithms_ub <- list(
68   "UBCF P 10" = list(name = "UBCF", param = list(nn = 10, method="pearson")),
69   "UBCF C 10" = list(name = "UBCF", param = list(nn = 10, method="cosine")),
70   "UBCF P 30" = list(name = "UBCF", param = list(nn = 30, method="pearson")),
71   "UBCF C 30" = list(name = "UBCF", param = list(nn = 30, method="cosine")),
72   "UBCF P 50" = list(name = "UBCF", param = list(nn = 50, method="pearson")),
73   "UBCF C 50" = list(name = "UBCF", param = list(nn = 50, method="cosine"))
74 )
75

```

Figura 24 - Exemplo da configuração para avaliação dos algoritmos

O esquema de avaliação descrito foi então aplicado às diferentes configurações, sendo que, para o método baseado em AR, fez-se variar o suporte e o grau de confiança e cuja nomenclatura reflete isso mesmo, ou seja, “AR 1 20” significa que a avaliação foi feita sobre o método baseado em regras associativas com um suporte de 1% e um grau de confiança de 20%.

Para o método IBCF e UBCF, fez-se variar a medida de similaridade utilizada e o número de vizinhos a considerar, sendo que “IBCF P 10” refere-se à utilização do método de IBCF utilizando o coeficiente de correlação de *Pearson* como medida de similaridade e dez vizinhos a considerar. “UBCF C 30” segue a mesma linha de pensamento, ou seja, refere-se à utilização do método UBCF com *Cosine Similarity* como medida de similaridade e trinta utilizadores vizinhos.

A partir dos resultados das avaliações foi possível traçar a curva ROC de cada algoritmo, bem como calcular a *F-Measure*. Estas duas métricas serão assim utilizadas para que seja identificada a configuração que tem os melhores resultados dentro de cada método de recomendação. Como analisado no capítulo 3.3.4.3, quanto mais alto for o valor da *F-Measure* melhores foram as recomendações dadas e, segundo a análise feita no Capítulo 3.3.4.4,

quanto mais próxima estiver a curva ROC do canto superior esquerdo, ou quanto maior for a área abaixo da curva, melhores foram as recomendações dadas.

Após identificação das configurações que obtiveram os melhores resultados nas avaliações, estas foram comparadas, mas agora em conjunto com os métodos de produtos aleatórios e produtos mais populares. O objetivo de toda esta análise é avaliar todos os métodos entre si, utilizando as configurações que melhores resultados têm, de forma a encontrar o método que apresenta os melhores resultados e, por consequência, entrará em produção.

5.2.4 Resultados da avaliação para AR

Para o método de recomendação baseado em regras associativas (AR) foram avaliadas seis configurações diferentes, onde se variou o grau de confiança entre 20%, 50% e 90% e o suporte, entre 1% e 2%, de forma a encontrar a configuração mais equilibrada. A partir destas avaliações foi possível a obtenção dos resultados apresentados no Anexo E – Resultados das avaliações para o método AR, de onde foi possível, a partir dos valores de precisão (*Precision*) e repetição (*Recall*) calcular a *F-Measure* e desenhar a curva ROC, através dos valores de TPR e FPR.

Para a primeira amostra, tendo em consideração os resultados da *F-Measure* presentes no Gráfico 5, é visível que a configuração com um suporte de 2% e uma confiança de 90% obteve os melhores resultados em conjunto com a configuração com um suporte de 2% e um grau de confiança de 20%.

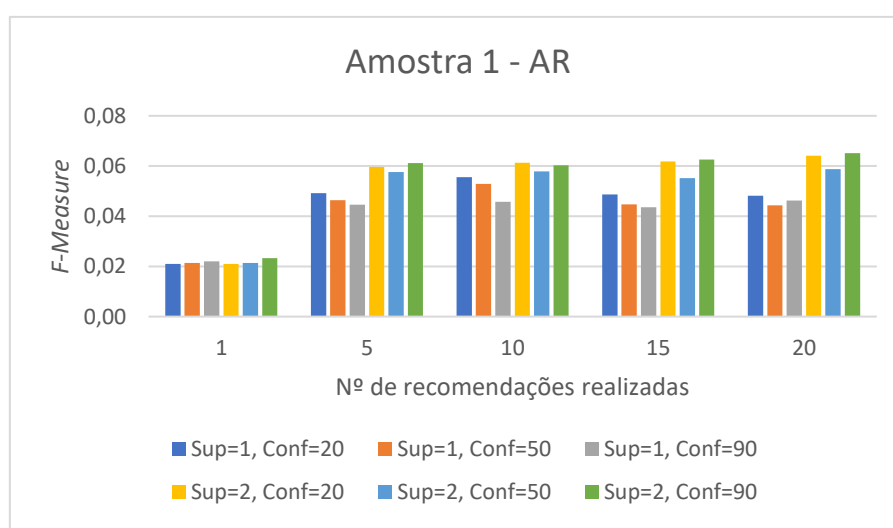


Gráfico 5 – Valores da *F-Measure* (Primeira amostra – AR)

Através da análise da curva ROC das configurações para a primeira amostra, traçada no Gráfico 6, verifica-se que existe uma diferença entre as taxas das recomendações positivas falsas

(FPR), onde se verifica uma FPR mais baixa para a configuração com um suporte de 2% e um grau de confiança de 90% em relação à configuração com um suporte de 2% e um grau de confiança de 20%, aumentando assim a área abaixo da curva desta última configuração. Com isto, conclui-se que a configuração com um suporte de 2% e um grau de confiança de 20% seja a que apresenta as melhores recomendações.

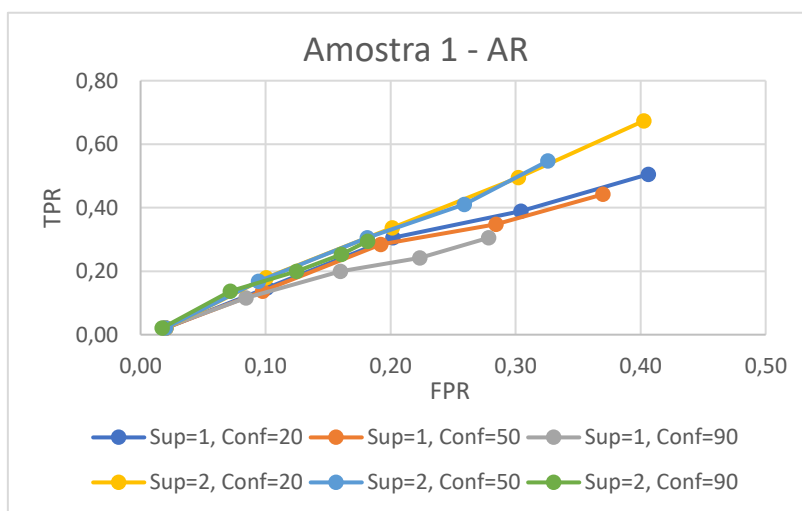


Gráfico 6 - Curva ROC (Primeira amostra - AR)

Para a segunda amostra, através da análise do Gráfico 7, pode verificar-se que, mais uma vez, as configuração com um suporte de 2% e um grau de confiança de 20% juntamente com a configuração com um suporte de 2% e um grau de confiança de 90% foram aquelas que obtiveram os resultados mais elevados. De notar que, para a recomendação de apenas um produto, as duas configurações com um suporte e grau de confiança mais baixos apresentam uma *F-Measure* igual a zero pois, através da análise da Tabela 18, presente no Anexo E – Resultados das avaliações para o método AR, nenhuma das configurações conseguiu recomendar um produto que um utilizador efetivamente já tivesse comprado.

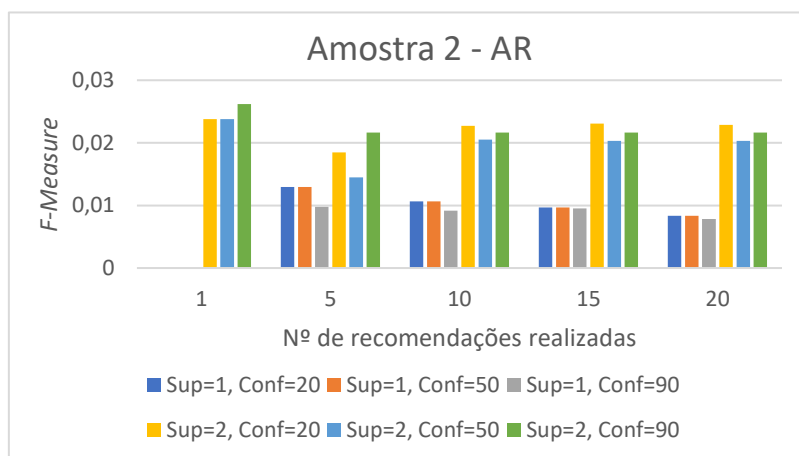


Gráfico 7 – Valores da *F-Measure* (Segunda amostra – AR)

Analisando as curvas ROC presentes no Gráfico 9, é possível verificar que, a curva referente à configuração com um suporte de 2% e uma confiança de 20% é a que mais próxima se encontra do canto superior esquerdo. Apesar de, em termos de área, as configurações com um suporte de 1% um grau de confiança de 20% e de 90% serem maiores, os seus resultados relativamente à *F-Measure* fazem com que as suas recomendações não sejam tão relevantes para o utilizador, fazendo com que, para esta segunda amostra, a configuração que melhor resultado obteve seja a que tem um suporte de 2% e um grau de confiança de 20%.

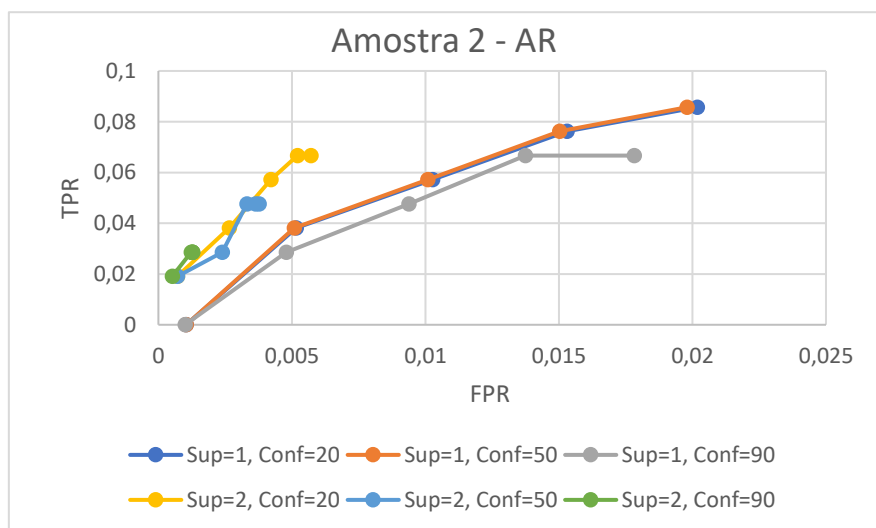


Gráfico 8 - Curva ROC (Segunda amostra - AR)

Para a terceira amostra, os resultados representados no Gráfico 9 não apresentam muitas semelhanças entre si, destacando-se, com os valores maiores, a configuração com um suporte de 1% e um grau de confiança de 20%.

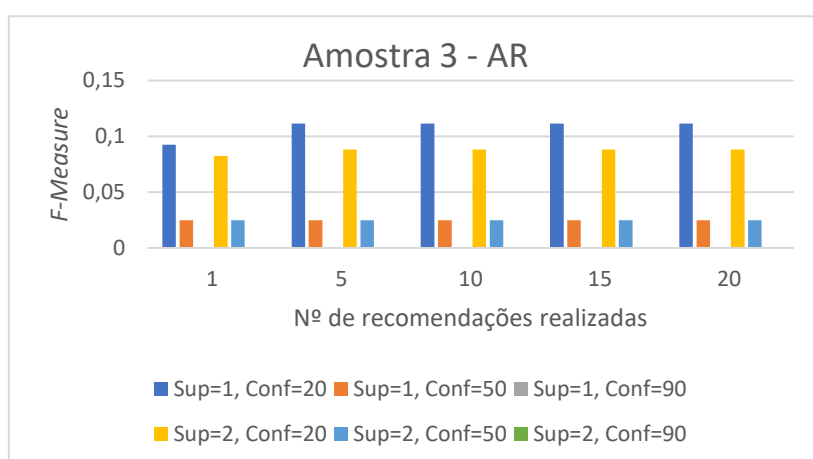


Gráfico 9 – Valores da F-Measure (Terceira amostra – AR)

Através da análise dos resultados foi possível identificar quais as configurações, visíveis na Tabela 10, realizaram as melhores recomendações. Através dessa análise foi também possível

verificar que, nas amostras onde a quantidade de produtos existentes no histórico dos utilizadores era menor, as configurações que melhores resultados obtiveram tinham também menores valores de suporte e confiança.

| | Nº Utilizadores | Nº Produtos | Suporte | Confiança |
|------------------|-----------------|-------------|---------|-----------|
| Amostra 1 | 87 | 48 | 2% | 20% |
| Amostra 2 | 101 | 944 | 2% | 20% |
| Amostra 3 | 676 | 98 | 1% | 20% |

Tabela 10 - Configurações mais relevantes para cada amostra do método AR

Com as características da amostra e os resultados obtidos, pode verificar-se que, para as amostras onde predominam compras feitas sobre os mesmos produtos, ou seja, existem muitas compras, mas sempre sobre o mesmo número de produtos, o suporte das regras encontradas será maior, porque estes aparecerão com mais frequência nas sequências de compras de outros utilizadores. Quando as compras se encontram dispersas pelos produtos, caso da terceira amostra, verificou-se que a configuração que obteve melhores resultados foi aquela que apresentou um suporte e grau de confiança mais baixos.

Desta forma, pode concluir-se que, dado o comportamento de compras dos utilizadores analisado no Capítulo 5.1, onde se verificou que existe uma frequência de compra relativamente elevada, ou seja, o número de compras da plataforma será elevado, uma configuração com um suporte de 2% e um grau de confiança de 20% será a configuração que realizará as recomendações mais adequadas aos utilizadores, e, por consequência, é aquela que melhor se adequa a entrar em produção.

5.2.5 Resultados da avaliação para IBCF

Para a avaliação das configurações de IBCF, fez-se uma análise de sensibilidade aos dois parâmetros seguintes:

- **k** – Quantidade de vizinhos a considerar para a composição da vizinhança dos produtos. Este parâmetro variou entre os valores 10, 30 e 50;
- **method** – Indica que medida de similaridade se pretende utilizar para encontrar os **k** produtos que constituem a vizinhança. A medida foi uma das seguintes: coeficiente de correlação de *Pearson*, ou *cosine similarity*.

Aplicando o esquema de avaliação apresentado no capítulo 5.2.3, foram obtidos os resultados presentes nas tabelas do Anexo F – Resultados das avaliações para o método IBCF. A partir

desse resultados foram traçadas as curvas ROC de cada avaliação, bem como calculada a *F-Measure*. Serão a partir dessas duas métricas que serão analisadas quais as configurações que melhor resultados obtiveram.

Através da análise do Gráfico 10, para a primeira amostra nenhuma configuração se destacou das restantes, mas a configuração com um k igual a 30 e que utiliza a *cosine similarity* como medida de similaridade é a que apresenta valores constantemente elevados em relação às restantes configurações.

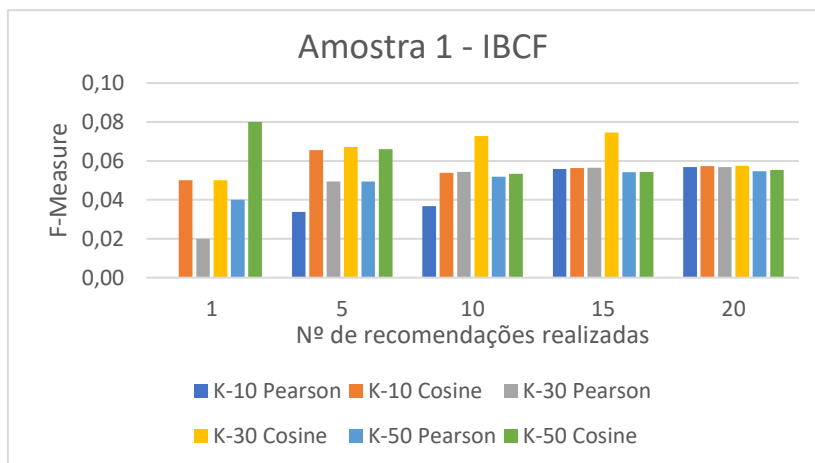


Gráfico 10 – Valores da *F-Measure* (Primeira amostra - IBCF)

Em conjunto com os valores da *F-Measure*, analisando as curvas ROC de cada avaliação, presentes no Gráfico 11, pode-se verificar que a que apresenta melhores resultados, quer pela proximidade da curva ao canto superior esquerdo, quer pela maior área abaixo da curva, é efetivamente a configuração que utiliza a *cosine similarity* como medida de similaridade e um k igual a 30.

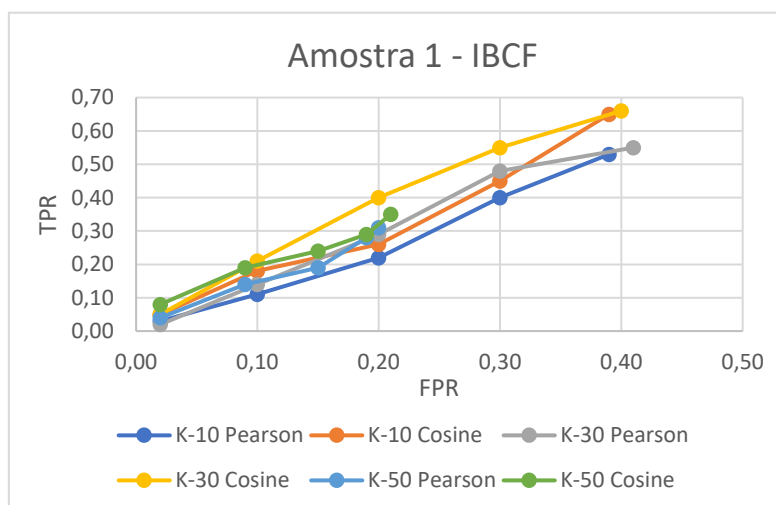


Gráfico 11 - Curva ROC (Primeira amostra - IBCF)

Para a segunda amostra, analisando o Gráfico 12, verifica-se uma configuração que se destaca das restantes, sendo esta a que tem um k igual a 10 e que utiliza o coeficiente de correlação de *Pearson* como medida de similaridade. Com este destaque em todas as avaliações, pode concluir-se que esta foi a configuração que, indubitavelmente, a configuração que realizou as recomendações mais relevantes.

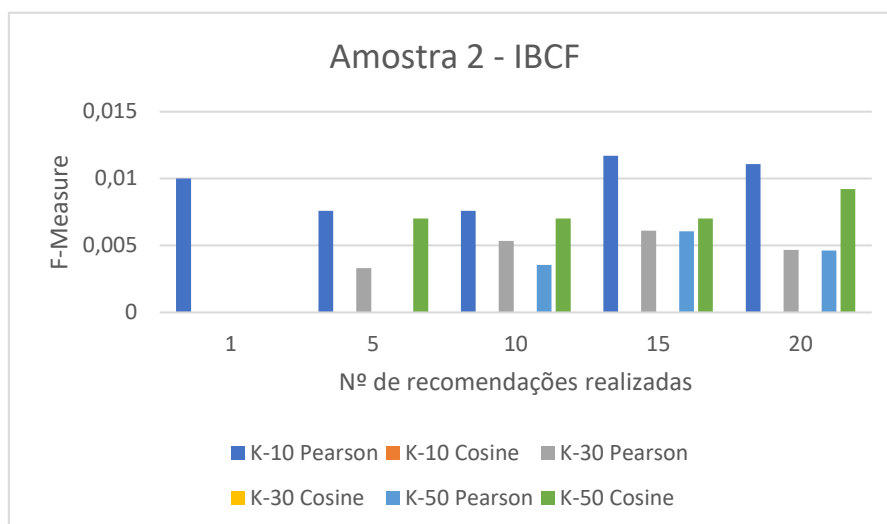


Gráfico 12 - Valores da F-Measure (Segunda amostra - IBCF)

Por fim, a partir dos dados evidenciados no Gráfico 13, não foi possível destacar uma configuração que tenha valores claramente mais elevados que as restantes, mas consegue-se verificar que a configuração com um k igual a 10 que utiliza uma *cosine similarity* como medida de similaridade tem uma ligeira vantagem sobre as restantes.

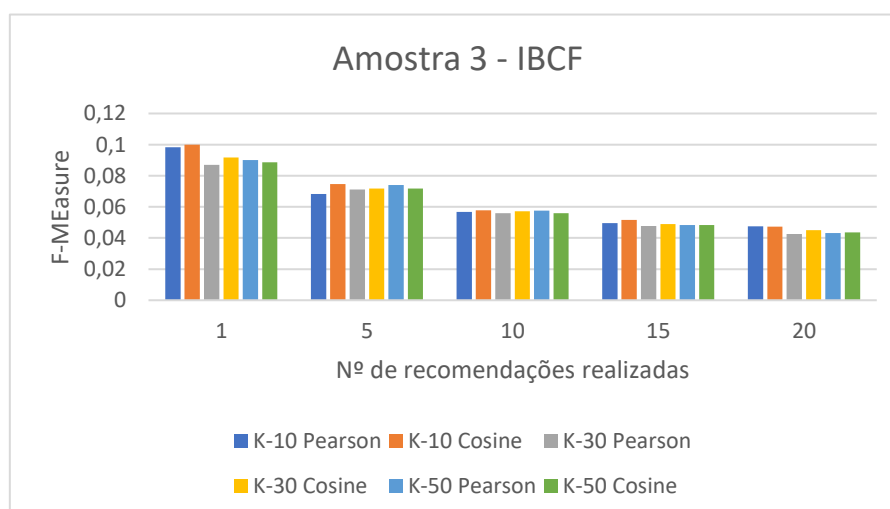


Gráfico 13 - Valores da F-Measure (Terceira amostra - IBCF)

Dados os resultados inconclusivos para a terceira amostra, será necessário proceder então à análise das respetivas curvas ROC das configurações. Analisando as curvas, presentes no

Gráfico 14, é possível verificar que, quer através da ligeira elevação comparativamente às restantes curvas, quer pela proximidade ao canto superior esquerdo, a curva da configuração com um k igual a 10 e que utiliza *cosine similarity* como medida de similaridade é aquela que melhores recomendações realizou. Assim, e em conjunto com a análise dos valores da *F-Measure* analisados anteriormente, conclui-se que esta é a configuração que, para a terceira amostra, melhores recomendações realiza.

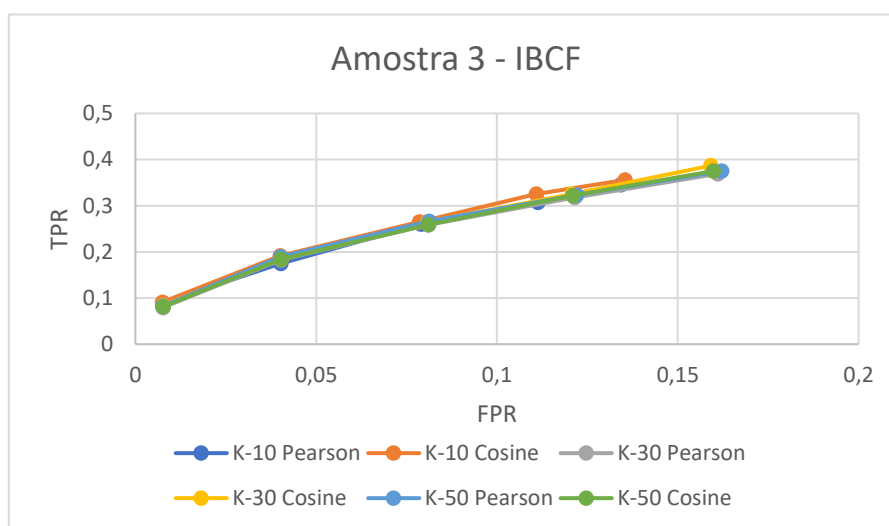


Gráfico 14 - Curva ROC (Terceira amostra - IBCF)

Com a análise dos resultados, resumidos na Tabela 11, não é possível definir uma relação direta entre o valor a utilizar como k, ou a medida de similaridade e as características dos dados utilizados como treino. Em contrapartida, pode verificar-se que o método IBCF obteve melhores classificações nas avaliações da primeira e terceira amostra. Isto pode indicar que o método IBCF terá melhores resultados quanto maior a frequência de compra de um produto. Por outras palavras, se os dados utilizados para treino contiveram demasiados produtos que foram comprados por poucos utilizadores, a eficácia do método IBCF será mais baixa, comparativamente à utilização de dados para treino que contenham produtos comprados por inúmeros utilizadores.

| | Nº Utilizadores | Nº Produtos | k | Medida Similaridade |
|------------------|-----------------|-------------|----|---------------------|
| Amostra 1 | 87 | 48 | 30 | <i>Cosine</i> |
| Amostra 2 | 101 | 944 | 10 | <i>Pearson</i> |
| Amostra 3 | 676 | 98 | 10 | <i>Cosine</i> |

Tabela 11 - Configurações mais relevantes para cada amostra do método IBCF

5.2.6 Resultados da avaliação para UBCF

De forma a identificar qual a configuração ideal a utilizar para o método de UBCF, foram avaliadas seis configurações onde se fizeram variar os seguintes parâmetros:

- **nn** – Quantidade de vizinhos a considerar para a composição da vizinhança dos utilizadores. Este parâmetro variou entre os valores 10, 30 e 50;
- **method** – Indica que medida de similaridade se pretende utilizar para encontrar os k produtos que constituem a vizinhança. A medida foi uma das seguintes: coeficiente de correlação de *Pearson*, ou *cosine similarity*.

Através dos resultados do Gráfico 15, é possível verificar que existem duas configurações que se destacam: nn igual a 10 com o coeficiente de correlação de *Pearson* e nn igual a 10 com *cosine similarity*. Dentro destas duas configurações é possível verificar que a primeira configuração apresenta valores de maiores grandezas.

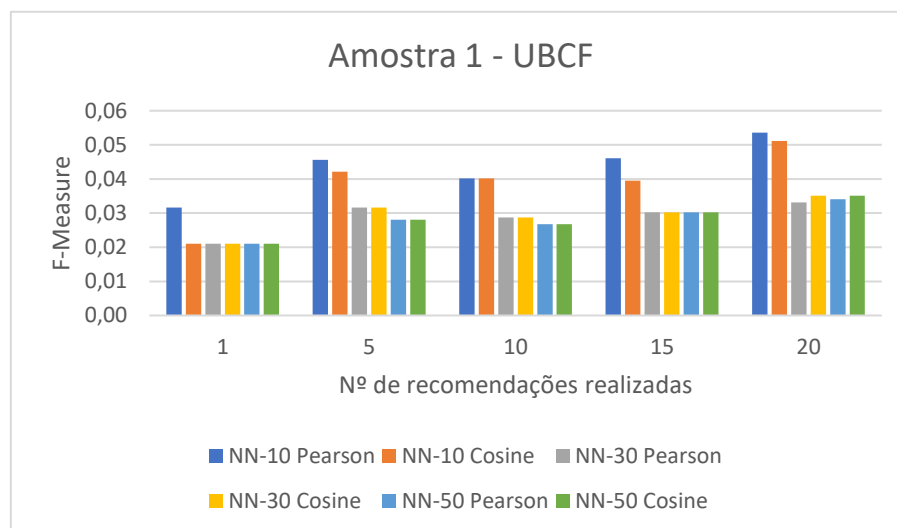


Gráfico 15 - Valores da F-Measure (Primeira amostra - UBCF)

Através da análise das curvas ROC das configurações, presentes no Gráfico 16, pode verificar-se que a configuração com um nn igual a 10 e que faz uso do coeficiente de relação de *Pearson* como medida de similaridade, além de ter uma maior área em relação às restantes curvas, é também aquela que mais se aproxima do canto superior esquerdo. Esta análise, em conjunto com os valores da *F-Measure* analisados anteriormente fazem desta configuração a que melhores recomendações realizou.

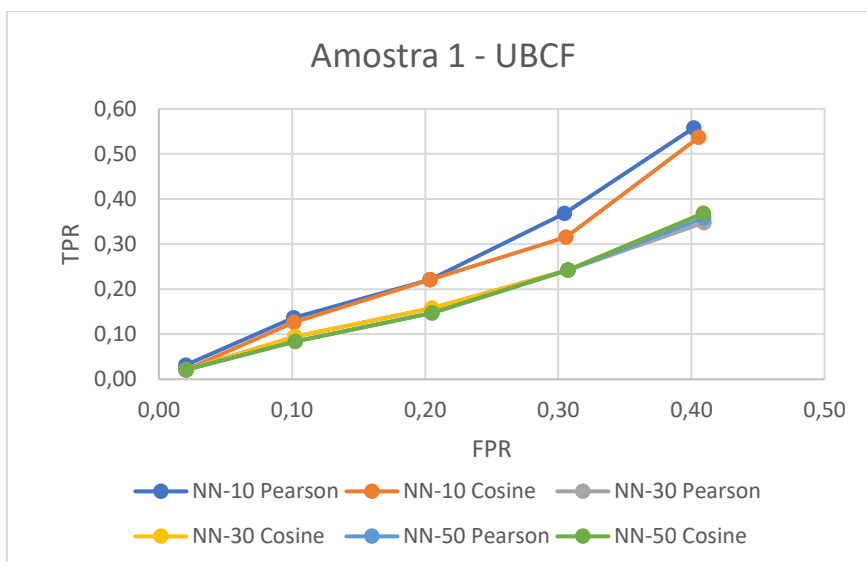


Gráfico 16 - Curva ROC (Primeira amostra - UBCF)

Através dos resultados presentes no Anexo G – Resultados das avaliações para o método UBCF pode-se verificar que nenhuma das configurações foi capaz de realizar uma lista com uma, ou cinco recomendações onde, pelo menos, uma delas fosse significativa para o utilizador. Além disso, pode-se verificar que as configurações que utilizam o coeficiente de correlação de *Pearson* como medida de similaridade não foram capazes de realizar recomendações tão significantes como as configurações que utilizam, o *cosine similarity* como medida de similaridade.

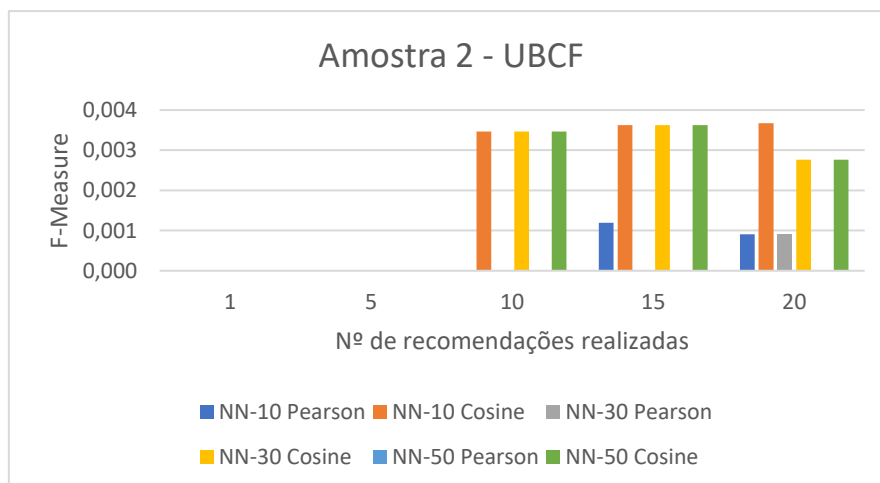


Gráfico 17 - Valores da F-Measure (Segunda amostra - UBCF)

Juntando aos valores da *F-Measure*, a análise das curvas ROC de cada configuração, presentes no Gráfico 18, pode-se verificar que a configuração com um nn igual a 10 e que utiliza *cosine similarity* tem uma curva a par com a configuração que tem um nn igual a 50 e também utiliza *cosine similarity*, mas, é na recomendação de vinte produtos que estas se diferenciam, dando

a vantagem para a primeira configuração e fazendo desta aquela que, mediante os resultados apresentados, é a que realiza as recomendações mais significantes.

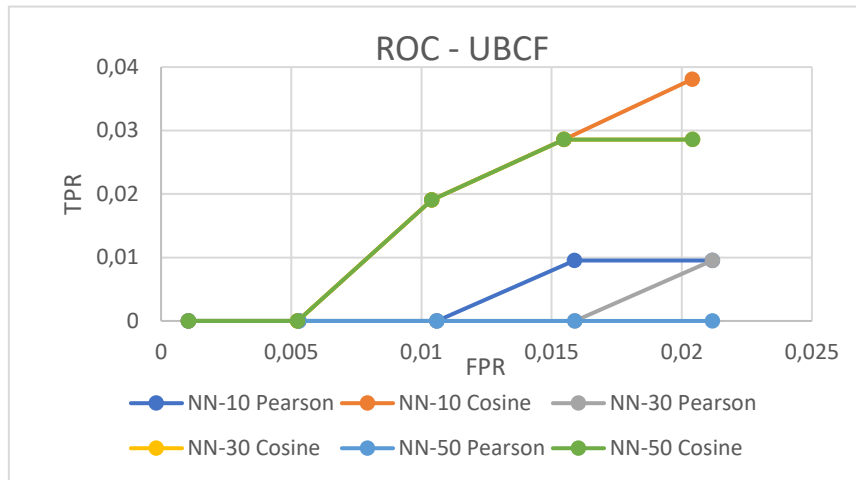


Gráfico 18 - Curva ROC (Segunda amostra - UBCF)

Por fim, pelo Gráfico 19, é possível verificar que se destacam as mesmas duas configurações: nn igual a 10 com o coeficiente de correlação de *Pearson* e nn igual a 10 com *cosine similarity*. Mas para esta amostra, não é possível destacar nenhuma destas configurações como sendo a que obteve melhores resultados e, por isso, terá de ser analisada a respetiva curva ROC.

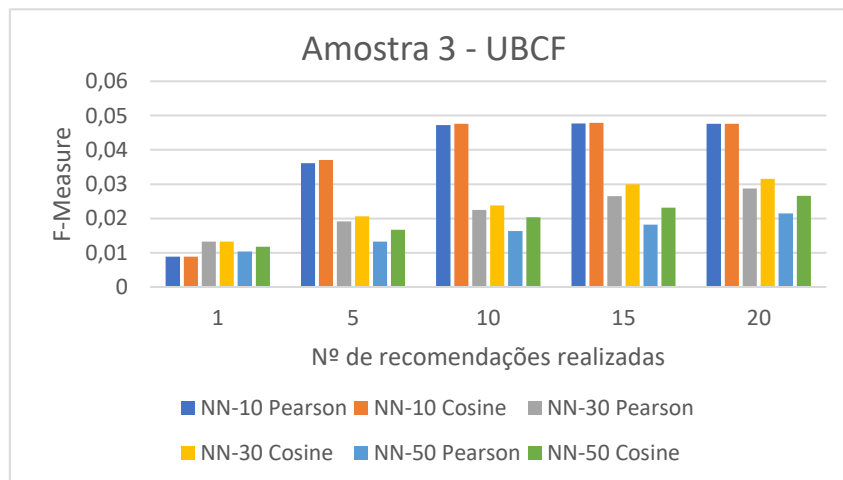


Gráfico 19 - Valores da F-Measure (Terceira amostra - UBCF)

Através da análise das curvas ROC, presentes no Gráfico 20, consegue-se verificar a similaridade entre as configurações indicadas anteriormente, mas, verifica-se a existência de uma ligeira vantagem para a curva da configuração que utiliza um nn igual a 10 e utilizando *cosine similarity* como medida de similaridade entre os utilizadores. É visível que esta curva, entre a recomendação de cinco e quinze produtos, apresenta uma pequena margem, apresentando uma maior área abaixo da sua curva.

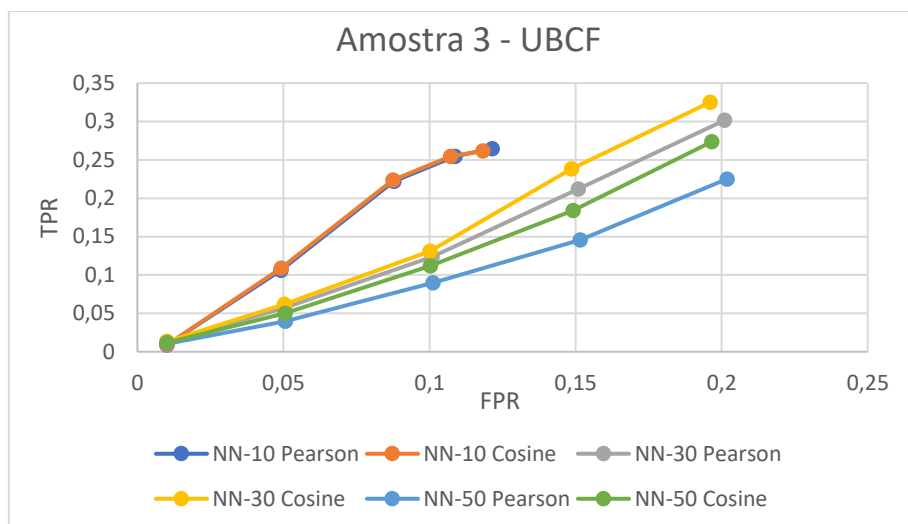


Gráfico 20 - Curva ROC (Terceira amostra - UBCF)

À semelhança do que foi analisado para os resultados das amostras do método IBCF, para o método UBCF também se verificaram resultados de maior grandeza para as amostras que tinham um histórico de utilizadores mais rico, ou seja, a primeira e terceira amostra. Dado que é utilizado o histórico dos utilizadores para identificar utilizadores com compras semelhantes entre si, quanto mais rico forem os históricos, mais eficazes serão os métodos.

Com isto, verifica-se que, para o método UBCF, as configurações que melhores recomendações realizaram, em cada amostra, se resumem da seguinte forma:

| | Nº Utilizadores | Nº Produtos | nn | Medida Similaridade |
|------------------|-----------------|-------------|----|---------------------|
| Amostra 1 | 87 | 48 | 10 | <i>Pearson</i> |
| Amostra 2 | 101 | 944 | 10 | <i>Cosine</i> |
| Amostra 3 | 676 | 98 | 10 | <i>Cosine</i> |

Tabela 12 - Configurações mais relevantes para cada amostra do método UBCF

5.2.7 Resultados da avaliação de todos os métodos de recomendação

A partir da análise dos resultados para cada configuração dentro de cada método, agora é necessário verificar como estas se comparam entre si. Para isso serão utilizadas as configurações identificadas anteriormente, resumidas na Tabela 13, em conjunto com os restantes métodos que não necessitam de configuração, nomeadamente: itens mais populares e itens aleatórios.

| | Método | Nº Utilizadores | Nº Produtos | Configuração |
|------------------|--------|-----------------|-------------|----------------------|
| Amostra 1 | AR | 87 | 48 | Sup=2%, Conf=20% |
| Amostra 2 | AR | 101 | 944 | Sup=2%, Conf=20% |
| Amostra 3 | AR | 676 | 98 | Sup=1%, Conf=20% |
| Amostra 1 | IBCF | 87 | 48 | k-30 <i>Cosine</i> |
| Amostra 2 | IBCF | 101 | 944 | k-10 <i>Pearson</i> |
| Amostra 3 | IBCF | 676 | 98 | k-10 <i>Cosine</i> |
| Amostra 1 | UBCF | 87 | 48 | nn-10 <i>Pearson</i> |
| Amostra 2 | UBCF | 101 | 944 | nn-10 <i>Cosine</i> |
| Amostra 3 | UBCF | 676 | 98 | nn-10 <i>Cosine</i> |

Tabela 13 - Configurações que melhores recomendações realizaram, em cada amostra

Através dos valores da *F-Measure*, presentes no Gráfico 21, verifica-se que as avaliações se encontram similares entre si, ou seja, não existe nenhum método que se destaque dos restantes. O único método que se destaca, mas pela negativa, é o método UBCF, pois é aquele que, no conjunto de todas as avaliações, apresenta os menores valores resultados da *F-Measure* em relação aos restantes.

Ainda assim, é possível verificar que os métodos baseados em AR e itens populares apresentam valores da *F-Measure* elevados, mas, ainda assim, no conjunto das avaliações feitas à primeira amostra, inferiores aos valores apresentados pelo método IBCF.

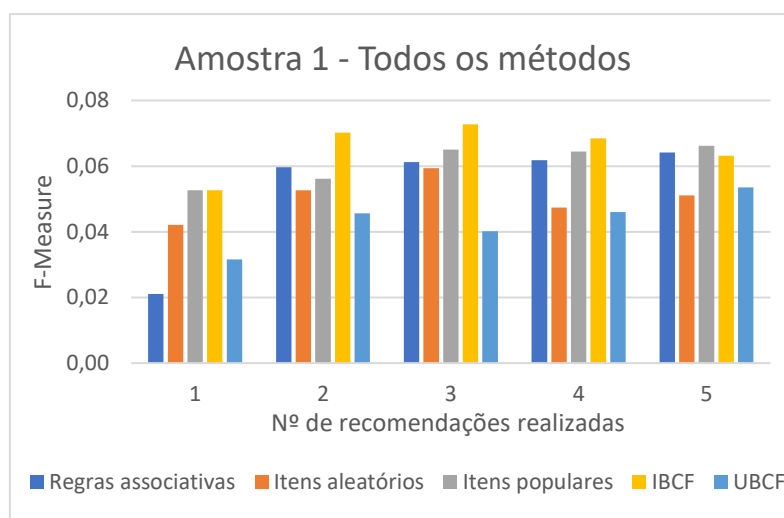


Gráfico 21 - Valores da *F-Measure* (Primeira amostra – todos os métodos)

No que diz respeito aos valores da *F-Measure* para a segunda amostra, ilustrados no Gráfico 22, verifica-se uma enorme vantagem para o método das regras associativas. Com esta vantagem, este método é, indiscutivelmente, o método que melhores recomendações realizou.

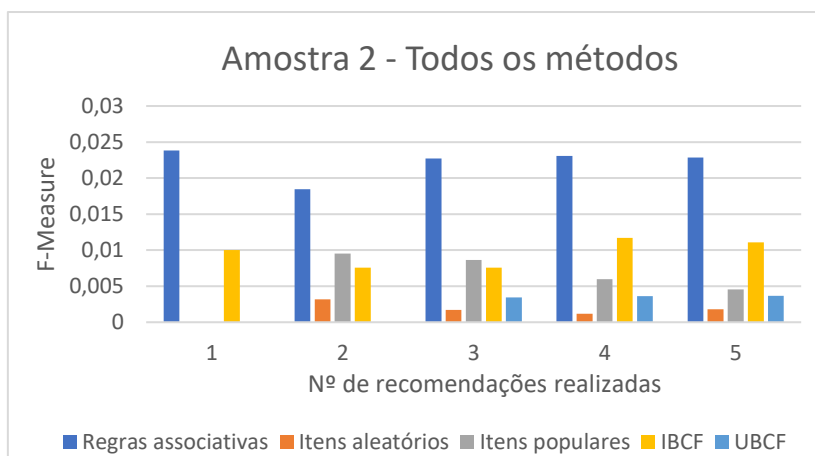


Gráfico 22 - Valores da *F-Measure* (Segunda amostra – todos os métodos)

Os valores da *F-Measure* para o método de itens mais populares baixou devido ao facto de cada produto que compõe a primeira amostra ser, em média, comprado catorze vezes, enquanto que, na segunda amostra, este valor diminui para duas compras por produto. Isto faz com a probabilidade de um utilizador comprar um produto popular seja menor, fazendo assim, com que os valores da *F-Measure* para este método baixem também.

Para a terceira amostra, na realização de apenas uma recomendação verificou-se, a partir dos resultados da *F-Measure* presentes no Gráfico 23, que o método IBCF apresenta melhores resultados que o de AR, mas, com o aumento do número de recomendações a realizar, o método IBCF vai tendo piores resultados e, em contrapartida, o método AR continua com os resultados elevados, destacando-se assim dos restantes e fazendo com que seja o método com os produtos recomendados mais prováveis de o utilizador comprar.

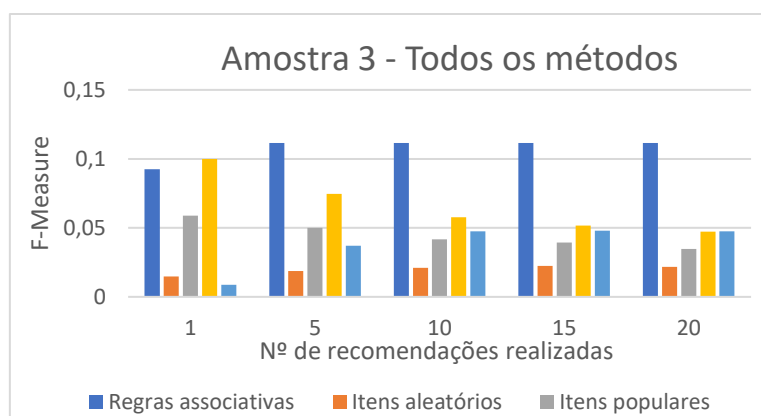


Gráfico 23 - Valores da *F-Measure* (Terceira amostra – todos os métodos)

Através da análise dos valores da *F-Measure* de todas as amostras foi possível verificar que um dos métodos que obteve os piores resultados foi o método de itens aleatórios, apenas com a exceção da primeira amostra. Este comportamento pode-se dever ao facto da primeira amostra ser constituída por poucos produtos que foram comprados muitas vezes (aproximadamente 14 compras por produto) aliado ao facto de a amostra conter poucos produtos (cerca de 48 produtos), ou seja, qualquer produto aleatório que seja recomendado, tem uma maior probabilidade de ser comprado pelo utilizador.

Por outro lado, o método de itens populares obteve resultados elevados, mas sempre com algum método a obter resultados mais elevados. Se for considerado que duas das amostras continham os produtos mais comprados e que este só mostrou maior precisão para a primeira amostra, pode indicar que este método só é indicado quando existem produtos onde as suas compras sejam bastante maiores que as dos restantes, ou onde os dados utilizados para treinar o sistema contenham poucos produtos.

Através desta análise, é possível verificar que cada método se comporta de maneira diferente mediante as características dos dados que compõem a matriz utilizada para o treino do modelo e isto faz com que, não exista um método, ou configuração, que possa ser utilizado para todos os clientes em produção. Cada cliente terá de ser analisado, à semelhança do trabalho realizado neste capítulo, de forma a que sejam identificados comportamentos de compras nos utilizadores e, de seguida, qual o método e, ou configuração que recomenda os produtos mais prováveis de um utilizador comprar e, com isso, entrará em produção para a solução do cliente.

Com isto, para o cliente analisado, é possível verificar que o método baseado em regras associativas teve um grande destaque nas avaliações feitas à segunda e terceira amostra, sendo que, mesmo para a primeira amostra, obteve resultados bastante elevados. Para a primeira amostra existem três métodos que se destacaram, sendo estes IBCF, itens populares e regras associativas, sequencialmente. Ainda assim, é possível verificar que o método baseado em regras associativas foi aquele que obteve melhores resultados considerando as três amostras, obtendo sempre uma maior precisão em todas elas e, mostrando, assim, ser o método mais viável para entrar em produção.

6 Implementação da solução

Neste capítulo será feita uma descrição, bem como justificativa das decisões tomadas da implementação do *Web Service*. Esta solução foi implementada em R, fazendo uso, principalmente, das bibliotecas *Recommenderlab*, para implementação e avaliação da lógica do mecanismo de recomendação e *plumber* para a implementação de uma API que disponibilizará os serviços da solução.

6.1 Método de recomendação

Como concluído na análise feita no Capítulo 5.2, para o cliente analisado, o método baseado em regras associativas foi aquele que melhores recomendações realizou, destacando-se na segunda e terceira amostras e mantendo resultados elevados para primeira amostra. Com isto, este método foi o que se verificou ser mais indicado para utilizar na solução, em conjunto com a utilização de um suporte de 2% e um grau de confiança de 20% devido à existência de uma frequência de compra relativamente elevada, ou seja, o número de compras da plataforma será elevado.

Apesar de este ser o método que realizou as melhores recomendações, o sistema fica com uma limitação que advém da sua utilização, que se trata do arranque a frio, ou *Cold Start*. A utilização deste método faz com que não seja possível a recomendação de produtos para um utilizador que ainda não tenha feito nenhuma compra.

Assim, foi analisada a possibilidade de implementação de um sistema de recomendação híbrido, através da utilização do método baseado em regras associativas em conjunto com o método de produtos mais populares. A escolha deste método deveu-se ao facto de, dentro dos métodos avaliados que conseguem ultrapassar a limitação, ser aquele que realizou as melhores recomendações.

A biblioteca utilizada apresenta uma solução de um sistema de recomendação híbrido, mas este não é de possível utilização uma vez que usa, para treinar o modelo, uma matriz de valores reais que não é compatível com a matriz utilizada para o método baseado em regras associativas, que apenas permite uma matriz de valores binários (Hahsler & Vereet, 2020). Este conflito é assim resolvido através da implementação de um mecanismo que visa a ultrapassar a limitação do arranque a frio. Este mecanismo recomenda produtos através da utilização do método baseado em regras associativas e, caso o utilizador não exista no sistema, utiliza o método de produtos mais populares para realizar as recomendações.

Com a implementação do mecanismo descrito, através do excerto de código da Figura 25, a limitação identificada fica ultrapassada.

```

1 # Importing libraries
2 library(data.table)
3 library(recommenderlab)
4 library(reshape2)
5
6 source("/app/DAL/MySQL/PurchasesDB.R"); #load database
7
8 trainModel <- function(){
9
10  usersHistory <- getAllPurchasesHistory() # Get all system purchases
11  historyParsed <- parseHistoryToRecommenderBinary(usersHistory) # Parse all purchases to a binaryRatingMatrix
12
13  arRecommender <- Recommender(historyParsed, method = "AR", param = list(supp = 0.02, conf = 0.2)) # Train AR recommender
14  popularRecommender <- Recommender(historyParsed, method = "POPULAR") # Train Popular recommender
15
16 }
17
18 parseHistoryToRecommenderBinary <- function(history){
19
20  history <- acast(history, UserId~ProductId, fill=FALSE)
21  return(as(history, "binaryRatingMatrix"))
22
23 }
24
25 recommendProductsToUser <- function(userId){
26
27  productsToRecommend <- 10
28
29  recommendations <- tryCatch({
30    userHistory <- historyParsed[tostring(userId)] # Get requested user purchases
31    recommendations <- predict(arRecommender, userHistory, n=productsToRecommend) # Predict with popular method
32  },
33  error=function(cond) { # Fallback if there is no AR recommendations
34    if( !exists("userHistory") ){
35      userHistory <- populateNonExistingUser() # get a matrix with no purchased product
36    }
37    recommendations <- predict(popularRecommender, userHistory, n=productsToRecommend) # Predict with popular method
38  })
39
40  return(as(recommendations, "list")) # return recommendations as a list
41
42 }
43
44 processUserPurchasesHistory <- function(purchases) {
45
46  return(insertUserPurchasesHistory(purchases))
47
48 }
49
50 populateNonExistingUser <- function() {
51
52  historyParsedMatrix <- as(historyParsed[1,], "matrix")
53
54  userMatrix <- rbind(historyParsedMatrix)
55  userMatrix <- replace(userMatrix, TRUE, FALSE)
56
57  return(as(userMatrix, "binaryRatingMatrix"))
58
59 }
60

```

Figura 25 - Código para o mecanismo de recomendação de produtos aos utilizadores

6.2 Arquitetura da solução

Foi tido em consideração o design apresentado no capítulo 4.2.2, onde é balanceada a complexidade do sistema com o seu desacoplamento da plataforma de E-commerce que fará uso das suas funcionalidades.



Figura 26 - Diagrama de componentes da solução

Para esse balanceamento foi então idealizada uma arquitetura cliente-servidor baseada em camadas, pelo que, através da Figura 26, encontram-se evidenciadas as três camadas principais: *Services*, *Models* e *DAL*.

A adoção de uma arquitetura por camadas potencializa a segurança, organização e escalabilidade do sistema, através da criação de uma hierarquia entre as camadas onde o acesso é restrito apenas às camadas adjacentes (Hanson, 2000).

Como indicado, para a implementação da lógica do algoritmo de recomendação foi utilizada a biblioteca *Recommenderlab*. Para disponibilizar as funcionalidades da solução através de uma API, foi utilizada a biblioteca *Plumber*. Desta forma, na Figura 27, encontram-se representadas, através de um diagrama de pacotes, todas as relações entre os elementos que compõem a solução e as bibliotecas utilizadas. Desta forma, é possível a esquematização das dependências existentes entre os elementos do sistema, bem como destes elementos e as bibliotecas externas.

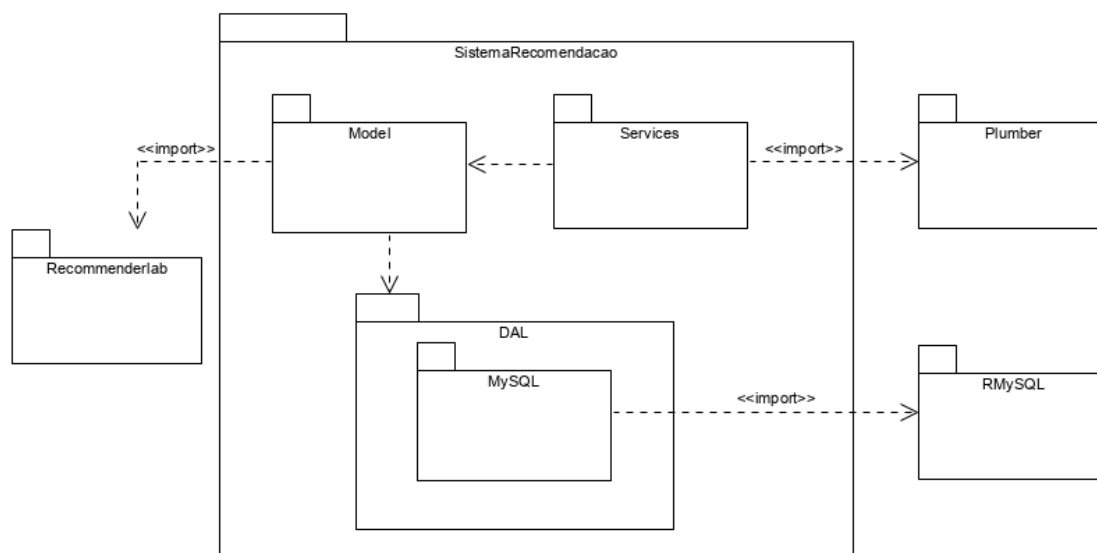


Figura 27 - Diagrama de pacotes da solução

6.2.1 Camada Services

Esta camada é responsável pela implementação do *Web Service* e disponibilização dos serviços à plataforma externa através de uma API REST (*Representational State Transfer*). Segundo Sharp (2011), um *Web Service* é um componente que fornece uma determinada funcionalidade reutilizável para o cliente, ou consumidor, podendo até ser pensado como um componente cuja acessibilidade seja verdadeiramente global. O *Web Service* é assim um *Web Service RESTful* que disponibiliza os serviços através de um URL (*Uniform Resource Location*) principal em conjunto com parâmetros definidos de acordo com o negócio. Por outras palavras, os serviços são disponibilizados através de um conjunto de parâmetros organizados de forma a que seja natural, para as regras do negócio, qual a finalidade do serviço que está a ser utilizado (Brito & Pereira, 2019).

Além do mencionado acima, existem outras regras, ou características que são necessárias para que o *Web Service* seja considerado *RESTful* (Brito & Pereira, 2019; Ribeiro, 2016):

- Ter como base uma arquitetura cliente-servidor;
- Não ter estado (*stateless*). Não pode ter conhecimento das aplicações que lhe estão conectadas nem possui conhecimento dos dados que estas aplicações possuem. Desta forma, trata cada execução de forma independente das restantes;
- Interface uniforme através da utilização de contratos entre o cliente e o servidor, onde se encontram bem definidos os acessos e regras para aceder aos serviços, de forma a tornar o sistema mais intuitivo.

Com a utilização da biblioteca *Plumber*, esta implementação fica mais facilitada, dado que esta biblioteca permite, através de uma notação específica nos comentários, disponibilizar código R através de serviços. Pode verificar-se, através do excerto de código presente na Figura 28, como é que os serviços da solução foram disponibilizados.

```

5  #* Train the model used to recommend products
6  #* @get /train
7  function(){
8
9      start.time <- Sys.time()
10     trainModel()
11     end.time <- Sys.time()
12
13     return(end.time - start.time)
14 }
15
16 # #* Get a list of recommendations to the given user
17 #* @get /recommend/<userId:int>
18 function(res, userId){
19
20     if(userId<=0){
21         res$status <- 404
22         list(error = "Invalid User")
23     }
24
25     return(recommendProductsToUser(userId))
26
27 }
28
29
30 # #* Insert a list of purchases into the DB
31 #* @serializer unboxedJSON
32 #* @post /purchases
33 function(req){
34
35     purchases <- as.data.frame(jsonlite::fromJSON(req$postBody))
36     process_result <- processUserPurchasesHistory(purchases)
37
38     return(process_result)
39
40 }
41

```

Figura 28 - Código para a disponibilização dos serviços da solução

6.2.2 Camada Models

A camada *Models* contém toda a lógica necessária para a identificação das recomendações, bem como regras de negócio que sejam necessárias para esta identificação. A biblioteca *Recommenderlab* permite a implementação de algoritmos tanto para a identificação das recomendações, como para verificar quais os utilizadores similares entre si.

Estes algoritmos, estariam assim contidos nesta camada. Desta forma, sempre que são pedidas ao *Web Service* as recomendações para um utilizador, é nesta camada que, através da aplicação do mecanismo mais pertinente, serão identificadas as recomendações.

6.2.3 Camada DAL

Toda a arquitetura foi idealizada tendo em conta a modularidade e desacoplamento entre as tecnologias e componentes da solução.

Para a base de dados da solução foi utilizado MySQL™. Esta escolha deveu-se principalmente ao facto de este ser o sistema de gestão de base de dados (SGBD) utilizado na empresa que, por consequência, é mais familiar para manutenção. Mas a solução torna este SGBD um

módulo, para o caso de haver a necessidade de ser substituído no futuro. Como exibido na Figura 26, a camada DAL é responsável por estabelecer a ligação com o SGBD e, com isso, obter os dados persistidos, bem como persistir dados. Dada esta abstração da implementação do SGBD, caso exista a necessidade de o mudar, bastará que seja implementado um novo módulo para comunicar com o novo SGBD e, utilizar esse módulo no sistema.

Caso haja alguma alteração mais drástica, como mudar a tecnologia do próprio *SistemaRecomendacoes*, dado que a plataforma de E-commerce comunica sempre para uma API, basta que o novo sistema disponibilize os mesmos serviços e, assim, a plataforma de E-commerce não tem de sofrer nenhuma alteração.

6.3 Funcionalidades da solução

Como indicado, a comunicação com a plataforma de E-commerce é feita através de pedidos HTTP ao *Web Service RESTful*. Todas as trocas de informação feitas entre a plataforma externa e o *Web Service* são realizadas através da utilização dos dados no formato JSON (JavaScript Object Notation).

As funcionalidades que a solução contém, descritas na Tabela 14, são então disponibilizadas através de um conjunto de serviços e acessíveis através de pedidos HTTP.

| | Método | URL | Descrição |
|-----------------------------------|--------|-------------------------------|--|
| Inserir compras no sistema | POST | {URL_BASE}/purchases | Recebe um número de compras e insere as mesmas no sistema |
| Treinar o modelo | GET | {URL_BASE}/train | Treina o modelo a utilizar pelo sistema de recomendação |
| Identificar recomendações | GET | {URL_BASE}/recommend/{UserId} | Dado um utilizador (UserId), identifica quais os produtos mais prováveis de este comprar |

Tabela 14 - Descrição das funcionalidades disponibilizadas pelo Web Service

6.3.1 Inserir compras no sistema

Para a inserção de compras no *SistemaRecomendacoes*, é recebida uma estrutura de dados igual à que se encontra na Figura 29 através de um pedido POST ao serviço acessível pelo parâmetro “/purchases” (ver exemplo de um pedido no Anexo C - Exemplo de um pedido cURL para inserir compras no sistema).

```

1  {
2  |   "Purchases": [
3  |     {
4  |       "OrderId": "1781374",
5  |       "UserId": "700000",
6  |       "CountryId": "1",
7  |       "Date": "01\08\2017",
8  |       "OrderValue": "25.97",
9  |       "OrderLineId": "10000000",
10 |       "ProductId": "11254",
11 |       "GenderDesc": "Man",
12 |       "ProductFamily": "4",
13 |       "ProductValue": "23.97"
14 |     },
15 |     {
16 |       "OrderId": "1781375",
17 |       "UserId": "1700",
18 |       "CountryId": "2",
19 |       "Date": "01\08\2017",
20 |       "OrderValue": "145.65",
21 |       "OrderLineId": "10000001",
22 |       "ProductId": "12345",
23 |       "GenderDesc": "Man",
24 |       "ProductFamily": "3",
25 |       "ProductValue": "59.95"
26 |     }
27 |   ]
28 | }

```

Figura 29 - Exemplo do corpo de um pedido para adicionar compras com duas compras

Para a construção da instrução que insere a compra na base de dados, foram implementados dois mecanismos de integração. O primeiro mecanismo concatena toda a informação necessária de forma direta, tornando o processo mais rápido, mas propenso a vulnerabilidades, tais como *SQL Injection*, que podem fazer com que, como o próprio nome indica, através da injeção de código SQL no corpo do pedido, causar danos na base de dados, ou obter dados que, de outra forma, seriam inacessíveis.

Foi esta vulnerabilidade que levou à implementação do segundo mecanismo, que faz uso de uma funcionalidade mais segura para tratar os dados a inserir, mas que, por outro lado, torna o processo mais lento.

| Nº de compras a inserir | 100 compras | 200 compras | 400 compras |
|--------------------------|-------------|-------------|-------------|
| Mecanismo 1 ⁵ | 877 ms | 2.06 s | 3.52 s |
| Mecanismo 2 ⁶ | 1.42 s | 2.32 s | 4.23 s |

Tabela 15 – Tempo necessário para inserir um conjunto de compras na base de dados

⁵ Mecanismo identificado como sendo o menos seguro, mas o mais rápido

⁶ Mecanismo identificado como sendo o mais seguro, mas mais lento

Desta forma, entendeu-se que, dado que a diferença entre os mecanismos não é muito grande e não existe nenhum requisito para o tempo máximo para inserção destas compras, optou-se por deixar o mecanismo 2 ativo. Desta forma a segurança encontra-se assegurada, mantendo o tempo de inserção das compras aceitável.

Com isto, na Figura 30 é possível visualizar o diagrama de sequência para a funcionalidade de inserir as compras dos utilizadores no sistema onde se verifica o comportamento da plataforma face à chegada de pedidos.

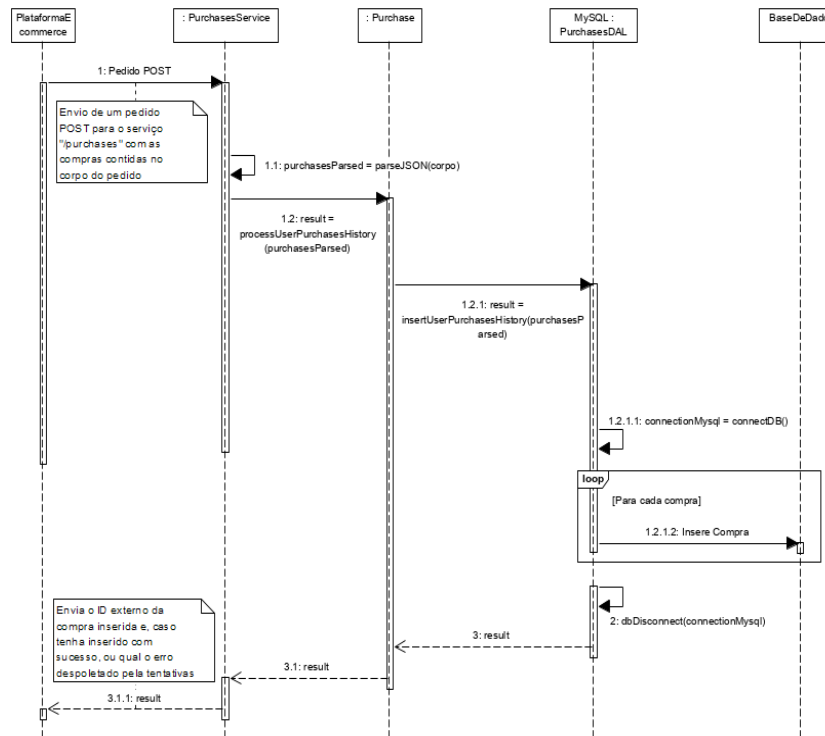


Figura 30 - Diagrama de sequência para a funcionalidade de inserir compras no sistema

6.3.2 Treinar o modelo

Dado que a transmissão dos dados não é constante, ou seja, não ocorre sempre a um determinado horário, o treino do modelo a utilizar para a recomendação de produtos tem de ser um mecanismo que seja flexível. Desta forma, foi disponibilizado um serviço, através do parâmetro “/train”, onde é despoletado o treino do modelo.

Um possível mecanismo para a atualização do modelo poderá passar pela atualização do modelo assim que atingido um determinado número de compras inseridas no sistema. Este mecanismo poderá fazer com que o modelo apenas seja atualizado caso tenham sido inseridas um número considerável de compras, mas, por outro lado, depende em demasia do fluxo de compras da plataforma, dado que, caso seja grande, o modelo poderá atualizar demasiadas

vezes num curto espaço de tempo e, por outro lado, caso o fluxo de compras seja pouco, poderá levar dias até atualizar novamente.

Dada esta inconsistência na inserção de compras e dado que não existe a necessidade, por parte do negócio, em manter o modelo atualizado regularmente, optou-se pela configuração de um Cron Job que despoletará o serviço de atualizar o modelo. Desta forma, o Cron Job é configurado para despoletar o serviço de quatro em quatro horas, mantendo o número de atualizações do modelo moderadas.

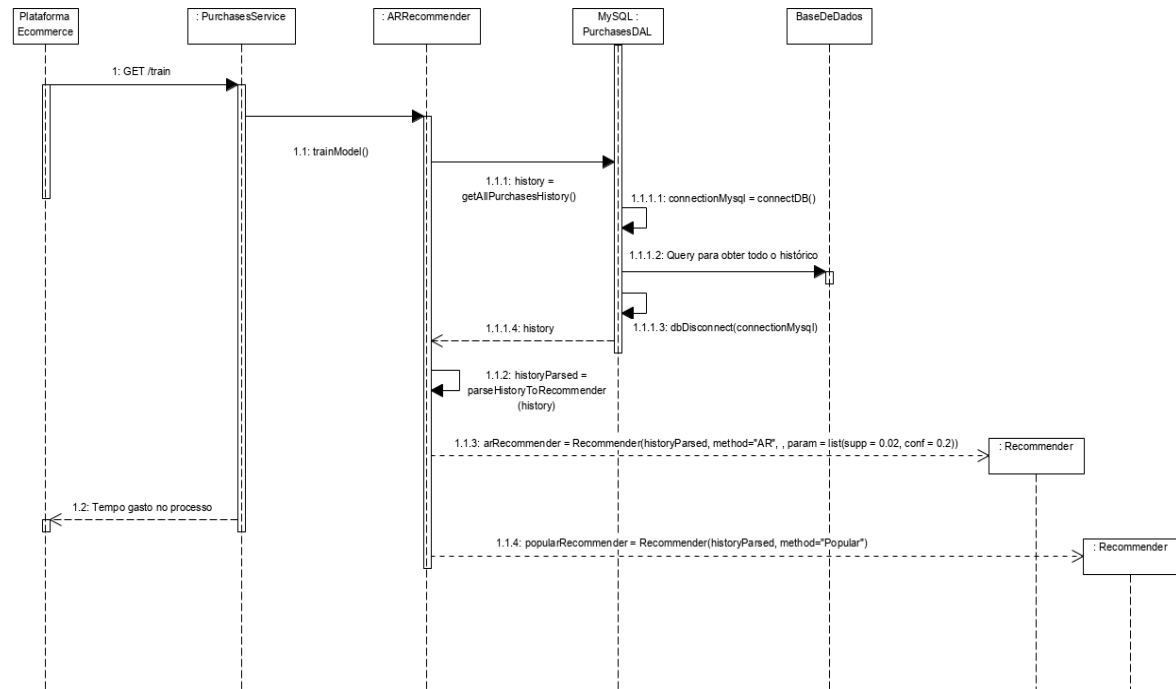


Figura 31 - Diagrama de sequência para a funcionalidade de treinar o modelo

Esta funcionalidade, além de treinar o modelo, disponibiliza globalmente uma variável com o histórico utilizado para o treino e o próprio objeto do modelo treinado para identificar as recomendações. Estas duas variáveis serão posteriormente utilizadas na funcionalidade de identificar recomendações.

6.3.3 Identificar recomendações

A funcionalidade de identificar recomendações, como o próprio nome indica, tem a função de, para um determinado utilizador, indicar quais os dez produtos mais prováveis de comprar. Esta funcionalidade encontra-se disponibilizada através do parâmetro “/recommend/{UserId}”, sendo que, {UserId} é o número identificador do utilizador para o sistema.

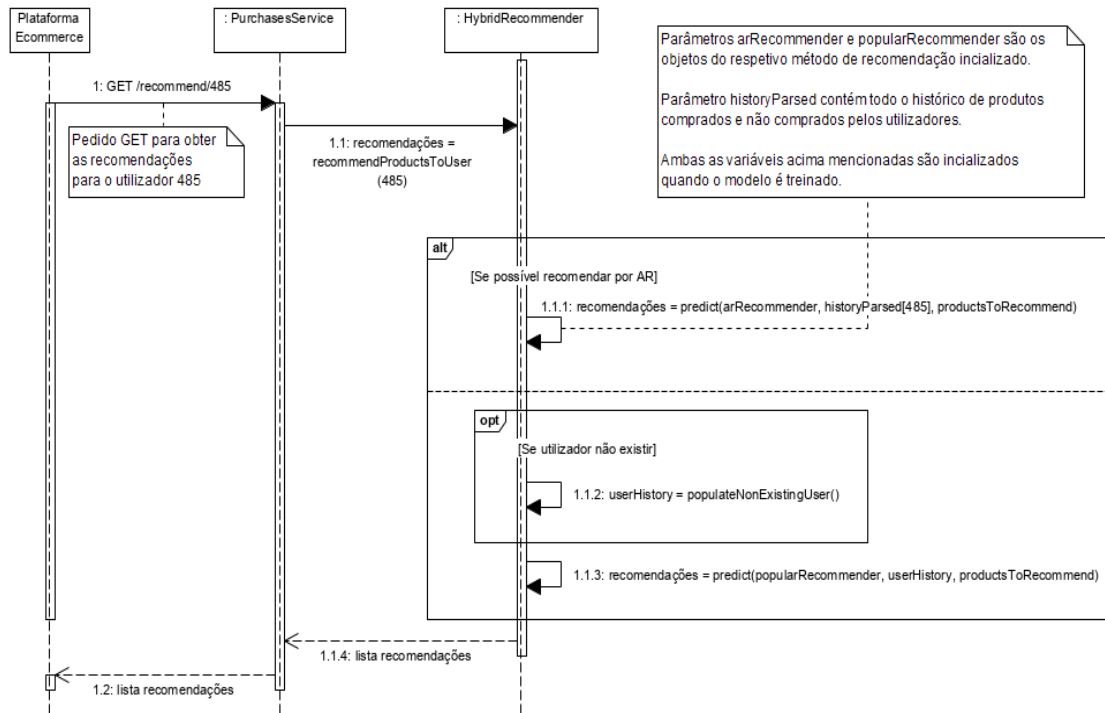


Figura 32 - Diagrama de sequência para a funcionalidade de identificar recomendações

6.4 Implantação da solução

Dado que a Redicom tem mais que um cliente, foi necessário ter essa característica em consideração de forma a que a solução possa facilmente ser disponibilizada para cada um deles. Desta forma, a solução pode ser colocada em produção em qualquer máquina através de um contentor Docker (Aruna et al., 2018).

Executar uma aplicação utilizando Docker permite a configuração e execução da solução num ambiente isolado, além de que, pode reduzir o atraso entre a escrita de código e a sua execução em ambiente de produção (Aruna et al., 2018).

Por exemplo, ao contrário da virtualização, contentores conseguem ser executados na mesma máquina partilhando o OS (sistema operativo) *kernel*, mas em processos isolados, ocupando menos espaço que máquinas virtuais (*Virtual Machines*) que têm de incluir uma cópia do OS, da aplicação em si e das bibliotecas e binários necessários para a execução. Estas diferenças encontram-se esquematizadas na Figura 33.

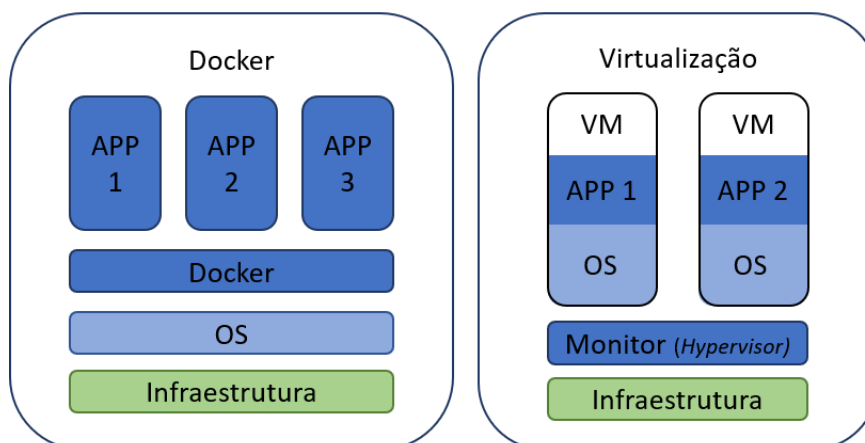


Figura 33 - Diferença entre Docker e virtualização⁷

Com esta solução é possível a execução de várias aplicações isoladas dentro do mesmo servidor, ou seja, a Redicom consegue ter um sistema de recomendação por cada cliente com a possibilidade de partilhar a mesma máquina e os respetivos recursos. Desta forma o *hardware* é maximizado de uma forma rápida e simples.

Todos os contentores têm como base uma imagem que é gerada através de ficheiros denominados de *Docker file*. Estas imagens são executáveis leves que incluem todo o código e dependências necessárias para que a aplicação seja executada de forma isolada (Aruna et al., 2018; *What Is a Container?*). Os *Docker files* utilizados para a execução da solução podem ser consultados no Anexo D - Ficheiros para execução da solução.

6.5 Requisitos satisfeitos

Como indicado na Tabela 8, existem requisitos funcionais aos quais a solução tem de responder. Abaixo encontra-se a lista com a identificação do requisito e a respetiva descrição a indicar se foi, ou não satisfeito.

- **REQF001** – Este requisito encontra-se cumprido dado que a plataforma disponibiliza um serviço onde, através do qual, é possível a inserção de experiências dos utilizadores;
- **REQF003** – Na solução encontra-se um algoritmo implementado que é capaz de, para um utilizador dado, responder com os produtos mais prováveis de este comprar, pelo que, o requisito pode ser dado como satisfeito;

⁷ Baseado em (*What Is a Container?*)

- **REQNF001** – A solução faz uso da biblioteca *plumber*. Esta biblioteca permite a implementação de uma API que faz uso do protocolo HTTP para as comunicações, fazendo com que este requisito esteja satisfeito;
- **REQF002, REQF004 e REQF005** – Dado que não existe nenhum algoritmo implementado que seja capaz de segmentar os clientes por grupos significativos, estes requisitos não se encontram satisfeitos na solução.

Com isto, pode-se verificar que os requisitos indicados pela empresa como sendo prioritários encontram-se satisfeitos, bem como as funcionalidades identificadas no capítulo da análise funcional como sendo as mais prioritárias, se encontram implementadas.

7 Conclusão

Em suma, podemos concluir que o sistema de recomendação foi identificado como sendo uma forma de resolver o problema existente e, efetivamente, foi o caso. Através da implementação desta solução foi possível, não só, automatizar um processo que é demasiado moroso para os clientes e propenso a falhas, como também foi possível desenvolver um sistema mais eficaz no que respeita às recomendações realizadas.

Para o cliente analisado, através das diferentes amostras e configurações avaliadas foi possível identificar o método e respetiva configuração que melhores resultados apresenta. A utilização do método baseado em regras associativas com um grau de suporte de 2% e uma confiança de 20% é a configuração mais adequada para o caso estudado. Ainda assim, foi identificada uma limitação deste método que passava pela falta de recomendações para utilizadores que sejam novos no sistema. Para ultrapassar esta limitação foi implementada uma solução que utiliza o método baseado em regras associativas para recomendar e, caso o utilizador não exista, faz uso do método baseado em produtos mais populares para realizar as recomendações.

Além disto, foi também verificado no Capítulo 5.2 como é que, mediante o comportamento de compras dos utilizadores, cada método se comporta face às recomendações identificadas. Apesar de não existir uma solução que sirva a todos os clientes, foi estruturada uma análise que poderá ser replicada aquando da aplicação desta solução em cada cliente. Esta análise passa pela caracterização dos hábitos de compra dos utilizadores que, em conjunto com avaliações a diferentes métodos de recomendação resultam na identificação do mecanismo que, para o cliente a analisar, recomenda os produtos de uma forma mais eficaz.

Em suma, com todas as avaliações e análises feitas, resultou uma compreensão do funcionamento dos métodos de recomendação, bem como uma experiência acerca de possíveis comportamentos e processos a serem aplicados de forma a que os produtos recomendados sejam os mais prováveis de um utilizador comprar.

7.1 Objetivos cumpridos

Neste projeto nem todos os objetivos foram cumpridos, nomeadamente os objetivos gerais:

- Aumento do número de transações na plataforma utilizando as recomendações dadas pela solução implementada;

- Aumento do número de cliques nos produtos recomendados pela solução implementada.

Estes objetivos não foram cumpridos uma vez que não foi possível a integração da solução na plataforma já existente de forma a avaliar se efetivamente se verificam um aumento do número de vendas e cliques proveniente dos produtos recomendados pela solução. A partir dos resultados obtidos nas avaliações, pode-se estimar que estes objetivos se iriam cumprir, mas, dado que não foi colocado em produção, não é possível esta suposição.

No que respeita aos objetivos específicos, estes foram todos cumpridos com sucesso:

- Implementação de um sistema de recomendação capaz de obter as informações pretendidas, a partir das experiências dos utilizadores;
- Automatização de processos de forma a otimizar processos no cliente.

Além de ter sido implementado um sistema de recomendação, devido à integração por *Web Service* com a plataforma existente e o automatismo do treino do modelo, não é necessária qualquer intervenção por parte dos clientes. Isto faz com que se deem por cumpridos estes dois objetivos.

7.2 Trabalhos futuros

Para o futuro, numa primeira instância poderá ser ajustado o sistema de recomendação para que seja contabilizado todas as experiências dos utilizadores. Uma possível solução para este ajuste pode passar pela utilização de uma matriz que, ao invés de ser binária, ou seja, contenha os valores 1, ou 0, caso o produto tenha sido comprado pelo utilizador, contenha classificações. Desta forma, quanto mais importante fosse a experiência, maior seria a classificação de um produto para um utilizador.

Outro possível ajuste seria a implementação de um algoritmo próprio. Por exemplo, um algoritmo que utilizasse o método de *clustering* para agrupar os utilizadores e, de seguida, identificasse as recomendações dentro de cada grupo encontrado. Este algoritmo poderia também encaminhar a solução para que fosse capaz de satisfazer os requisitos **REQF002**, **REQF004** e **REQF005**.

Outra sugestão poderia passar pelo melhoramento da performance das recomendações pedidas, dado que estas poderiam ficar guardadas em cache, por um sistema de base de dados em memória, por exemplo, o Redis (*Introduction to Redis – Redis*). Poderia ser feito de forma a que, quando o modelo fosse treinado seria gerada uma chave e quando fosse pedido

recomendações para um determinado utilizador, o sistema primeiro iria ver se para o utilizador com a chave mais recente tem dados guardados, caso tivesse, retornava esses dados, caso contrário, iria identificar as recomendações, guardar as mesmas em cache e devolvê-las.

Por fim, a última sugestão passa pela avaliação da solução atual em produção. Esta avaliação pode ser realizada através de testes A/B de forma a verificar se a configuração que se encontra implementada é a ideal, ou necessita de algum ajuste.

Bibliografia

- Aditya, P. H., Budi, I., & Munajat, Q. (2016). A comparative analysis of memory-based and model-based collaborative filtering on the implementation of recommender system for E-commerce in Indonesia: A case study PT X. *2016 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, 303–308. <https://doi.org/10.1109/ICACSIS.2016.7872755>
- Adomavicius, G., & Tuzhilin, A. (2005). *Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions*. 734–749. <https://doi.org/10.1109/TKDE.2005.99>
- Aruna, P., Kannammal, A., Aishwarya, & Surya. (2018). Distributed Data Analysis with Docker Swarm Using R. *Proceedings of the 2018 International Conference on Computational Intelligence and Intelligent Systems - CIIS 2018*, 10–13. <https://doi.org/10.1145/3293475.3293492>
- Balabanović, M., & Shoham, Y. (1997). Fab: Content-based, collaborative recommendation. *Communications of the ACM*, 40(3), 66–72. <https://doi.org/10.1145/245108.245124>
- Bartolomei, J. E., & Miller, T. (2001). *Functional Analysis Systems Technique (F.A.S.T.) as a Group Knowledge Elicitation Method for Model Building*. 10.
- Beel, J., Genzmehr, M., Langer, S., Nürnberger, A., & Gipp, B. (2013). A comparative analysis of offline and online evaluations and discussion of research paper recommender system evaluation. *Proceedings of the International Workshop on Reproducibility and Replication in Recommender Systems Evaluation - RepSys '13*, 7–14. <https://doi.org/10.1145/2532508.2532511>
- Breese, J. S., & Kadie, D. H. C. (1998). *Empirical Analysis of Predictive Algorithms for Collaborative Filtering*. 43–52.

- Brito, J. N., & Pereira, J. L. (2019). From a Desktop Application to a Web API A Code Conversion Approach. *Proceedings of the 2019 2nd International Conference on Geoinformatics and Data Analysis*, 6–10. <https://doi.org/10.1145/3318236.3318253>
- Davis, J., & Goadrich, M. (2006). The relationship between Precision-Recall and ROC curves. *Proceedings of the 23rd International Conference on Machine Learning - ICML '06*, 233–240. <https://doi.org/10.1145/1143844.1143874>
- Dobbelaere, P., & Esmaili, K. S. (2017). Kafka versus RabbitMQ: A comparative study of two industry reference publish/subscribe implementations: Industry Paper. *Proceedings of the 11th ACM International Conference on Distributed and Event-Based Systems - DEBS '17*, 227–238. <https://doi.org/10.1145/3093742.3093908>
- DRM Associates. (2016). *Value Analysis and Function Analysis System Technique*. 10.
- ENACHE, M.-C. (2019). *Machine Learning in E-commerce*. <https://doi.org/10.26397/eai1584040950>
- Funcionamento do sistema de recomendações da Netflix*. (2019). Centro de Assistência. <https://help.netflix.com/pt-pt/node/100639>
- Gilotte, A., Calauzènes, C., Nedelec, T., Abraham, A., & Dollé, S. (2018). Offline A/B Testing for Recommender Systems. *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining - WSDM '18*, 198–206. <https://doi.org/10.1145/3159652.3159687>
- Hahsler, M. (2020). *recommenderlab: A Framework for Developing and Testing Recommendation Algorithms*. 40.
- Hahsler, M., & Vereet, B. (2020). *Lab for Developing and Testing Recommender Algorithms*.
- Hanson, M. (2000). *The Client/Server Architecture*. <https://doi.org/10.1201/9781420031065.sec1>

- Herlocker, J., Konstan, J., Terveen, L., & Riedl, J. (2004). *Evaluating collaborative filtering recommender systems*. 22, 5–53. <https://doi.org/10.1145/963770.963772>
- Hu, Y., Koren, Y., & Volinsky, C. (2008). *Collaborative Filtering for Implicit Feedback Datasets*. 263–272. <https://doi.org/10.1109/ICDM.2008.22>
- Hwangbo, H., Kim, Y. S., & Cha, K. J. (2018). Recommendation system development for fashion retail e-commerce. *Electronic Commerce Research and Applications*, 28, 94–101. <https://doi.org/10.1016/j.elerap.2018.01.012>
- Infosys. (2013). *Rethinking Retail*.
- Introduction to Redis – Redis*. (sem data). Obtido 4 de Setembro de 2020, de <https://redis.io/topics/introduction>
- Isinkaye, F. O., Folajimi, Y. O., & Ojokoh, B. A. (2015). Recommendation systems: Principles, methods and evaluation. *Egyptian Informatics Journal*, 16(3), 261–273. <https://doi.org/10.1016/j.eij.2015.06.005>
- Jeunen, O. (2019). Revisiting offline evaluation for implicit-feedback recommender systems. *Proceedings of the 13th ACM Conference on Recommender Systems - RecSys '19*, 596–600. <https://doi.org/10.1145/3298689.3347069>
- Karypi, G. (2001). *Evaluation of Item-Based Top-N Recommendation Algorithms*. 247–254. <https://doi.org/10.1145/502585.502627>
- Khandagale, S., Mallade, S., Kharat, K., & Bansode, V. (2016). *Food Recommendation System Using Sequential Pattern Mining*. 2(6), 4.
- Koen, P., Ajamian, G., Burkart, R., Clamen, A., Davidson, J., D'Amore, R., Elkins, C., Herald, K., Incorvia, M., Johnson, A., Karol, R., Seibert, R., Slavejkov, A., & Wagner, K. (2001). Providing Clarity and A Common Language to the “Fuzzy Front End”. *Research-*

Technology Management, 44(2), 46–55.

<https://doi.org/10.1080/08956308.2001.11671418>

Konstan, J. A., Miller, B. N., Maltz, D., Herlocker, J. L., Gordon, L. R., & Riedl, J. (1997).

GroupLens: Applying collaborative filtering to Usenet news. *Communications of the ACM*, 40(3), 77–87. <https://doi.org/10.1145/245108.245126>

Kumar, M. V. (2019). A Study on Different Phases and Various Recommendation System Techniques. *Soft Computing*, 7(5), 4.

Patel, A. A., & Dharwa, J. N. (2016). Fuzzy Based Hybrid Mobile Recommendation System.

Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies - ICTCS '16, 1–6. <https://doi.org/10.1145/2905055.2905205>

Perez, L. G., Chiclana, F., & Ahmadi, S. (2011). A social network representation for Collaborative Filtering Recommender Systems. *2011 11th International Conference on Intelligent Systems Design and Applications*, 438–443.

<https://doi.org/10.1109/ISDA.2011.6121695>

Qin, Z., Shundong, L., Yi, H., Jinchun, D., Lixiang, Y., & Jun, Q. (2009). *Introduction to E-commerce*. Springer.

Raza, S., & Ding, C. (2019). *Progress in context-aware recommender systems—An overview*.

<https://doi.org/10.1016/j.cosrev.2019.01.001>

Ribeiro, M. F. (2016). *Conceitos, análise e implementação*. 18.

Saaty, T. L. (1980). *The Analytic Hierarchy Process*.

Sarwar, B., Karypis, G., Konstan, J., & Reidl, J. (2001). Item-based collaborative filtering recommendation algorithms. *Proceedings of the Tenth International Conference on World Wide Web - WWW '01*, 285–295. <https://doi.org/10.1145/371920.372071>

- Sarwar, B. M., Konstan, J. A., Borchers, A., Herlocker, J., Miller, B., & Riedl, J. (1998). Using filtering agents to improve prediction quality in the GroupLens research collaborative filtering system. *Proceedings of the 1998 ACM Conference on Computer Supported Cooperative Work - CSCW '98*, 345–354. <https://doi.org/10.1145/289444.289509>
- Schafer, J. B., Konstan, J., & Riedl, J. (1999). *Recommender Systems in E-Commerce*. 9.
- Sharp, J. (2011). *MICROSOFT VISUAL C# 2010 PASSO A PASSO*. Bookman.
- Shen, E., Lieberman, H., & Lam, F. (2007). What am I gonna wear?: Scenario-oriented recommendation. *Proceedings of the 12th International Conference on Intelligent User Interfaces - IUI '07*, 365. <https://doi.org/10.1145/1216295.1216368>
- So, W. T., & Yada, K. (2017). A Framework of Recommendation System Based on In-store Behavior. *Proceedings of the 4th Multidisciplinary International Social Networks Conference on ZZZ - MISNC '17*, 1–4. <https://doi.org/10.1145/3092090.3092130>
- Vaidya, N., & Khachane, A. R. (2017). Recommender systems-the need of the ecommerce ERA. *2017 International Conference on Computing Methodologies and Communication (ICCMC)*, 100–104. <https://doi.org/10.1109/ICCMC.2017.8282616>
- Welcome to Apache PredictionIO®! (sem data). Obtido 5 de Janeiro de 2020, de <https://predictionio.apache.org/>
- What is a Container? | App Containerization | Docker. (sem data). Obtido 6 de Setembro de 2020, de <https://www.docker.com/resources/what-container>
- Why TensorFlow | TensorFlow. (sem data). Obtido 5 de Janeiro de 2020, de <https://www.tensorflow.org/about?hl=pt>

Anexo

Anexo A - Escala fundamental de Saaty

| Nível de importância | Definição | Explicação |
|----------------------|-------------------------|---|
| 1 | Igual importância | As duas atividades contribuem igualmente para o objetivo |
| 3 | Fraca importância | A experiência e o julgamento favorecem levemente uma atividade em relação à outra |
| 5 | Forte importância | A experiência e o julgamento favorecem fortemente uma atividade em relação à outra |
| 7 | Muito forte importância | Uma atividade é muito fortemente favorecida em relação a outra |
| 9 | Importância absoluta | A evidência favorece uma atividade em relação a outra com o mais alto grau de certeza |
| 2,4,6,8 | Valores intermediários | Quando se procura uma condição de compromisso entre duas definições |

Tabela 16 – Escala fundamental de Saaty com os níveis de importância de comparações⁸

⁸ (Saaty, 1980)

Anexo B - Cálculo do valor de RC

Primeiro terá de ser encontrado o valor de λ_{max} , para que depois seja utilizado no cálculo do IC. Desta forma, λ_{max} poderá ser obtido através da seguinte fórmula: $Ax = \lambda_{max} x$, sendo que A é a matriz de critérios e x é o vetor próprio.

$$\begin{bmatrix} 1 & 3 & 4 \\ 0.33 & 1 & 2 \\ 0.25 & 0.5 & 1 \end{bmatrix} \begin{bmatrix} 0.6232 \\ 0.2395 \\ 0.1373 \end{bmatrix} = \lambda_{max} \begin{bmatrix} 0.6232 \\ 0.2395 \\ 0.1373 \end{bmatrix} \Leftrightarrow$$

$$\begin{bmatrix} 1.8908 \\ 0.7218 \\ 0.4128 \end{bmatrix} = \lambda_{max} \begin{bmatrix} 0.6232 \\ 0.2395 \\ 0.1373 \end{bmatrix} \Leftrightarrow$$

$$\lambda_{max} = \begin{bmatrix} 3.03 \\ 3.01 \\ 3.01 \end{bmatrix}$$

Com esta matriz, pode-se calcular λ_{max} como sendo a média dos valores da matriz calculada acima, ou seja, $\lambda_{max} \simeq 3.02$. Com este valor é então possível calcular o valor de IC através da fórmula (1).

$$IC = \frac{\lambda_{max} - n}{n - 1} \Leftrightarrow IC = \frac{3.02 - 3}{3 - 1} \Leftrightarrow IC \simeq 0.01$$

A juntar este valor e, através da análise do valor de IR para um $n=3$ na Tabela 4, ou seja, 0.58, pode-se calcular então o RC a partir da fórmula (2).

$$RC = \frac{IC}{IR} \Leftrightarrow RC = \frac{0.01}{0.58} \Leftrightarrow RC \simeq 0.02$$

Anexo C - Exemplo de um pedido cURL para inserir compras no sistema

```
1  curl --location --request POST 'http://192.168.99.100:8001/purchases' \  
2  --header 'Content-Type: application/json' \  
3  --data-raw '{  
4      "Purchases": [  
5          {  
6              "OrderId": "1781374",  
7              "UserId": "700000",  
8              "CountryId": "1",  
9              "Date": "01\08\2017",  
10             "OrderValue": "25.97",  
11             "OrderLineId": "1000000",  
12             "ProductId": "11254",  
13             "GenderDesc": "Man",  
14             "ProductFamily": "4",  
15             "ProductValue": "23.97"  
16         },  
17         {  
18             "OrderId": "1781375",  
19             "UserId": "1700",  
20             "CountryId": "2",  
21             "Date": "01\08\2017",  
22             "OrderValue": "145.65",  
23             "OrderLineId": "1000001",  
24             "ProductId": "12345",  
25             "GenderDesc": "Man",  
26             "ProductFamily": "3",  
27             "ProductValue": "59.95"  
28         }  
29     ]  
30 }
```

Figura 34 - Pedido cURL para inserir compras no sistema

Anexo D - Ficheiros para execução da solução

Ficheiro *docker-compose* com a configuração para a execução da solução:

```
version : '3'

services:
  mysql:
    image: mysql:latest
    container_name: rec_mysql
    environment:
      MYSQL_USER: user
      MYSQL_PASSWORD: user
      MYSQL_ROOT_USER: root
      MYSQL_ROOT_PASSWORD: root
      MYSQL_DATABASE: default_schema
    ports:
      - 3306:3306
  phpmyadmin:
    image: phpmyadmin/phpmyadmin
    container_name: rec_mysql_pma
    links:
      - mysql
    environment:
      PMA_HOST: mysql
      PMA_PORT: 3306
      PMA_ARBITRARY: 1
    restart: always
    ports:
      - 8002:80
  api:
    build: ./app/
    command: /app/Services/PurchasesService.R
    container_name: rec_api
    depends_on:
      - mysql
    links:
      - mysql
    volumes:
      - ./app:/app
    restart: always
    ports:
      - 8001:8000
```

Ficheiro docker com a configuração para a execução da aplicação R:

```
# start from the rocker/r-ver:3.5.0 image
FROM rocker/r-base

# install plumber
RUN R -e
"install.packages(c('RMySQL','plumber','data.table','reshape2','reco
mmenderlab'))"

RUN apt-get update -qq && apt-get install -y \
  git-core \
  libssl-dev \
  libcurl4-gnutls-dev \
  default-libmysqlclient-dev

RUN R -e
"install.packages(c('RMySQL','data.table','reshape2','recommenderlab
'))"

ENTRYPOINT ["R", "-e", "pr <- plumber::plumb(commandArgs()[4]);
pr$run(host='0.0.0.0', port=8000)"]
```

Anexo E – Resultados das avaliações para o método AR

| Nº recomendações | Precision | Recall | TPR | FPR | F-Measure |
|------------------|-----------|----------|----------|----------|-----------|
| AR 1 20 | | | | | |
| 1 | 0.021053 | 0.021053 | 0.021053 | 0.020395 | 0.02 |
| 5 | 0.029474 | 0.147368 | 0.147368 | 0.101096 | 0.05 |
| 10 | 0.030526 | 0.305263 | 0.305263 | 0.201974 | 0.06 |
| 15 | 0.025965 | 0.389474 | 0.389474 | 0.304386 | 0.05 |
| 20 | 0.025263 | 0.505263 | 0.505263 | 0.40614 | 0.05 |
| AR 1 50 | | | | | |
| 1 | 0.021637 | 0.021053 | 0.021053 | 0.019956 | 0.02 |
| 5 | 0.027953 | 0.136842 | 0.136842 | 0.097588 | 0.05 |
| 10 | 0.029123 | 0.284211 | 0.284211 | 0.192105 | 0.05 |
| 15 | 0.023904 | 0.347368 | 0.347368 | 0.284211 | 0.04 |
| 20 | 0.023324 | 0.442105 | 0.442105 | 0.369956 | 0.04 |
| AR 1 90 | | | | | |
| 1 | 0.023026 | 0.021053 | 0.021053 | 0.01864 | 0.02 |
| 5 | 0.027617 | 0.115789 | 0.115789 | 0.08443 | 0.04 |
| 10 | 0.025828 | 0.2 | 0.2 | 0.159868 | 0.05 |
| 15 | 0.023935 | 0.242105 | 0.242105 | 0.223465 | 0.04 |
| 20 | 0.025022 | 0.305263 | 0.305263 | 0.278509 | 0.05 |
| AR 2 20 | | | | | |
| 1 | 0.021053 | 0.021053 | 0.021053 | 0.020395 | 0.02 |
| 5 | 0.035789 | 0.178947 | 0.178947 | 0.100439 | 0.06 |
| 10 | 0.033684 | 0.336842 | 0.336842 | 0.201316 | 0.06 |
| 15 | 0.032982 | 0.494737 | 0.494737 | 0.302193 | 0.06 |
| 20 | 0.033684 | 0.673684 | 0.673684 | 0.402632 | 0.06 |
| AR 2 50 | | | | | |
| 1 | 0.021637 | 0.021053 | 0.021053 | 0.019956 | 0.02 |
| 5 | 0.034737 | 0.168421 | 0.168421 | 0.094298 | 0.06 |
| 10 | 0.03193 | 0.305263 | 0.305263 | 0.18136 | 0.06 |
| 15 | 0.029552 | 0.410526 | 0.410526 | 0.258991 | 0.06 |
| 20 | 0.031065 | 0.547368 | 0.547368 | 0.325877 | 0.06 |
| AR 2 90 | | | | | |
| 1 | 0.025911 | 0.021053 | 0.021053 | 0.017325 | 0.02 |
| 5 | 0.039396 | 0.136842 | 0.136842 | 0.071711 | 0.06 |
| 10 | 0.035487 | 0.2 | 0.2 | 0.124781 | 0.06 |
| 15 | 0.035697 | 0.252632 | 0.252632 | 0.160746 | 0.06 |
| 20 | 0.036642 | 0.294737 | 0.294737 | 0.181579 | 0.07 |

Tabela 17 - Resultados das avaliações ao método AR para a primeira amostra

| Nº recomendações | Precision | Recall | TPR | FPR | F-Measure |
|------------------|-----------|----------|----------|----------|-----------|
| AR 1 20 | | | | | |
| 1 | 0 | 0 | 0 | 0.001039 | 0 |
| 5 | 0.00781 | 0.038095 | 0.038095 | 0.005155 | 0.0129619 |
| 10 | 0.005857 | 0.057143 | 0.057143 | 0.01026 | 0.0106252 |
| 15 | 0.005175 | 0.07619 | 0.07619 | 0.015295 | 0.009691 |
| 20 | 0.004381 | 0.085714 | 0.085714 | 0.020178 | 0.0083359 |
| AR 1 50 | | | | | |
| 1 | 0 | 0 | 0 | 0.001039 | 0 |
| 5 | 0.00781 | 0.038095 | 0.038095 | 0.005085 | 0.0129619 |
| 10 | 0.005857 | 0.057143 | 0.057143 | 0.010089 | 0.0106252 |
| 15 | 0.005175 | 0.07619 | 0.07619 | 0.015022 | 0.009691 |
| 20 | 0.004381 | 0.085714 | 0.085714 | 0.019794 | 0.0083359 |
| AR 1 90 | | | | | |
| 1 | 0 | 0 | 0 | 0.000999 | 0 |
| 5 | 0.005905 | 0.028571 | 0.028571 | 0.004792 | 0.0097869 |
| 10 | 0.005074 | 0.047619 | 0.047619 | 0.009383 | 0.0091714 |
| 15 | 0.005141 | 0.066667 | 0.066667 | 0.013741 | 0.0095462 |
| 20 | 0.004148 | 0.066667 | 0.066667 | 0.017817 | 0.0078106 |
| AR 2 20 | | | | | |
| 1 | 0.031765 | 0.019048 | 0.019048 | 0.000706 | 0.0238148 |
| 5 | 0.012186 | 0.038095 | 0.038095 | 0.002653 | 0.0184656 |
| 10 | 0.014186 | 0.057143 | 0.057143 | 0.004217 | 0.0227297 |
| 15 | 0.013942 | 0.066667 | 0.066667 | 0.005206 | 0.0230608 |
| 20 | 0.013806 | 0.066667 | 0.066667 | 0.00571 | 0.0228752 |
| AR 2 50 | | | | | |
| 1 | 0.031765 | 0.019048 | 0.019048 | 0.000706 | 0.0238148 |
| 5 | 0.009686 | 0.028571 | 0.028571 | 0.002391 | 0.0144677 |
| 10 | 0.013075 | 0.047619 | 0.047619 | 0.003309 | 0.0205168 |
| 15 | 0.012893 | 0.047619 | 0.047619 | 0.003642 | 0.0202923 |
| 20 | 0.012893 | 0.047619 | 0.047619 | 0.003773 | 0.0202923 |
| AR 2 90 | | | | | |
| 1 | 0.041905 | 0.019048 | 0.019048 | 0.000515 | 0.0261905 |
| 5 | 0.017403 | 0.028571 | 0.028571 | 0.001231 | 0.0216303 |
| 10 | 0.017403 | 0.028571 | 0.028571 | 0.001271 | 0.0216303 |
| 15 | 0.017403 | 0.028571 | 0.028571 | 0.001271 | 0.0216303 |
| 20 | 0.017403 | 0.028571 | 0.028571 | 0.001271 | 0.0216303 |

Tabela 18 - Resultados das avaliações ao método AR para a segunda amostra

| Nº recomendações | Precision | Recall | TPR | FPR | F-Measure |
|------------------|-----------|----------|----------|----------|-----------|
| AR 1 20 | | | | | |
| 1 | 0.146522 | 0.067647 | 0.067647 | 0.004067 | 0.09256 |
| 5 | 0.136707 | 0.094118 | 0.094118 | 0.007593 | 0.111483 |
| 10 | 0.136487 | 0.094118 | 0.094118 | 0.007683 | 0.11141 |
| 15 | 0.136487 | 0.094118 | 0.094118 | 0.007683 | 0.11141 |
| 20 | 0.136487 | 0.094118 | 0.094118 | 0.007683 | 0.11141 |
| AR 1 50 | | | | | |
| 1 | 0.183586 | 0.013235 | 0.013235 | 0.000585 | 0.024691 |
| 5 | 0.183586 | 0.013235 | 0.013235 | 0.000585 | 0.024691 |
| 10 | 0.183586 | 0.013235 | 0.013235 | 0.000585 | 0.024691 |
| 15 | 0.183586 | 0.013235 | 0.013235 | 0.000585 | 0.024691 |
| 20 | 0.183586 | 0.013235 | 0.013235 | 0.000585 | 0.024691 |
| AR 1 90 | | | | | |
| 1 | 0 | 0 | 0 | 0 | 0 |
| 5 | 0 | 0 | 0 | 0 | 0 |
| 10 | 0 | 0 | 0 | 0 | 0 |
| 15 | 0 | 0 | 0 | 0 | 0 |
| 20 | 0 | 0 | 0 | 0 | 0 |
| AR 2 20 | | | | | |
| 1 | 0.205602 | 0.051471 | 0.051471 | 0.002026 | 0.082331 |
| 5 | 0.210112 | 0.055882 | 0.055882 | 0.002581 | 0.088284 |
| 10 | 0.210112 | 0.055882 | 0.055882 | 0.002581 | 0.088284 |
| 15 | 0.210112 | 0.055882 | 0.055882 | 0.002581 | 0.088284 |
| 20 | 0.210112 | 0.055882 | 0.055882 | 0.002581 | 0.088284 |
| AR 2 50 | | | | | |
| 1 | 0.183586 | 0.013235 | 0.013235 | 0.000585 | 0.024691 |
| 5 | 0.183586 | 0.013235 | 0.013235 | 0.000585 | 0.024691 |
| 10 | 0.183586 | 0.013235 | 0.013235 | 0.000585 | 0.024691 |
| 15 | 0.183586 | 0.013235 | 0.013235 | 0.000585 | 0.024691 |
| 20 | 0.183586 | 0.013235 | 0.013235 | 0.000585 | 0.024691 |
| AR 2 90 | | | | | |
| 1 | 0 | 0 | 0 | 0 | 0 |
| 5 | 0 | 0 | 0 | 0 | 0 |
| 10 | 0 | 0 | 0 | 0 | 0 |
| 15 | 0 | 0 | 0 | 0 | 0 |
| 20 | 0 | 0 | 0 | 0 | 0 |

Tabela 19 - Resultados das avaliações ao método AR para a terceira amostra

Anexo F – Resultados das avaliações para o método IBCF

| Nº recomendações | Precision | Recall | TPR | FPR | F-Measure |
|---------------------|-----------|--------|------|------|-----------|
| K-10 Pearson | | | | | |
| 1 | 0.03 | 0.03 | 0.03 | 0.02 | 0.00 |
| 5 | 0.02 | 0.11 | 0.11 | 0.1 | 0.03 |
| 10 | 0.02 | 0.22 | 0.22 | 0.2 | 0.04 |
| 15 | 0.03 | 0.4 | 0.4 | 0.3 | 0.06 |
| 20 | 0.03 | 0.53 | 0.53 | 0.39 | 0.06 |
| K-10 Cosine | | | | | |
| 1 | 0.05 | 0.05 | 0.05 | 0.02 | 0.05 |
| 5 | 0.04 | 0.18 | 0.18 | 0.1 | 0.07 |
| 10 | 0.03 | 0.26 | 0.26 | 0.2 | 0.05 |
| 15 | 0.03 | 0.45 | 0.45 | 0.3 | 0.06 |
| 20 | 0.03 | 0.65 | 0.65 | 0.39 | 0.06 |
| K-30 Pearson | | | | | |
| 1 | 0.02 | 0.02 | 0.02 | 0.02 | 0.02 |
| 5 | 0.03 | 0.14 | 0.14 | 0.1 | 0.05 |
| 10 | 0.03 | 0.29 | 0.29 | 0.2 | 0.05 |
| 15 | 0.03 | 0.48 | 0.48 | 0.3 | 0.06 |
| 20 | 0.03 | 0.55 | 0.55 | 0.41 | 0.06 |
| K-30 Cosine | | | | | |
| 1 | 0.05 | 0.05 | 0.05 | 0.02 | 0.05 |
| 5 | 0.04 | 0.21 | 0.21 | 0.1 | 0.07 |
| 10 | 0.04 | 0.4 | 0.4 | 0.2 | 0.07 |
| 15 | 0.04 | 0.55 | 0.55 | 0.3 | 0.07 |
| 20 | 0.03 | 0.66 | 0.66 | 0.4 | 0.06 |
| K-50 Pearson | | | | | |
| 1 | 0.04 | 0.04 | 0.04 | 0.02 | 0.04 |
| 5 | 0.03 | 0.14 | 0.14 | 0.09 | 0.05 |
| 10 | 0.03 | 0.19 | 0.19 | 0.15 | 0.05 |
| 15 | 0.03 | 0.28 | 0.28 | 0.19 | 0.05 |
| 20 | 0.03 | 0.31 | 0.31 | 0.2 | 0.05 |
| K-50 Cosine | | | | | |
| 1 | 0.08 | 0.08 | 0.08 | 0.02 | 0.08 |
| 5 | 0.04 | 0.19 | 0.19 | 0.09 | 0.07 |
| 10 | 0.03 | 0.24 | 0.24 | 0.15 | 0.05 |
| 15 | 0.03 | 0.29 | 0.29 | 0.19 | 0.05 |
| 20 | 0.03 | 0.35 | 0.35 | 0.21 | 0.06 |

Tabela 20 - Resultados das avaliações ao método IBCF para a primeira amostra

| Nº recomendações | Precision | Recall | TPR | FPR | F-Measure |
|---------------------|-----------|----------|----------|----------|-----------|
| K-10 Pearson | | | | | |
| 1 | 0.010526 | 0.009524 | 0.009524 | 0.000989 | 0.01 |
| 5 | 0.004737 | 0.019048 | 0.019048 | 0.004863 | 0.0075869 |
| 10 | 0.004737 | 0.019048 | 0.019048 | 0.009463 | 0.0075869 |
| 15 | 0.006673 | 0.047619 | 0.047619 | 0.0136 | 0.0117062 |
| 20 | 0.00627 | 0.047619 | 0.047619 | 0.017464 | 0.0110807 |
| K-10 Cosine | | | | | |
| 1 | 0 | 0 | 0 | 0.000454 | 0 |
| 5 | 0 | 0 | 0 | 0.002179 | 0 |
| 10 | 0 | 0 | 0 | 0.004157 | 0 |
| 15 | 0 | 0 | 0 | 0.005993 | 0 |
| 20 | 0 | 0 | 0 | 0.007809 | 0 |
| K-30 Pearson | | | | | |
| 1 | 0 | 0 | 0 | 0.001039 | 0 |
| 5 | 0.002 | 0.009524 | 0.009524 | 0.005186 | 0.0033058 |
| 10 | 0.002952 | 0.028571 | 0.028571 | 0.010291 | 0.0053517 |
| 15 | 0.00327 | 0.047619 | 0.047619 | 0.015305 | 0.0061195 |
| 20 | 0.002452 | 0.047619 | 0.047619 | 0.020167 | 0.0046645 |
| K-30 Cosine | | | | | |
| 1 | 0 | 0 | 0 | 0.000767 | 0 |
| 5 | 0 | 0 | 0 | 0.003723 | 0 |
| 10 | 0 | 0 | 0 | 0.007375 | 0 |
| 15 | 0 | 0 | 0 | 0.010835 | 0 |
| 20 | 0 | 0 | 0 | 0.014266 | 0 |
| K-50 Pearson | | | | | |
| 1 | 0 | 0 | 0 | 0.001039 | 0 |
| 5 | 0 | 0 | 0 | 0.005196 | 0 |
| 10 | 0.001952 | 0.019048 | 0.019048 | 0.010301 | 0.0035417 |
| 15 | 0.003238 | 0.047619 | 0.047619 | 0.015325 | 0.0060638 |
| 20 | 0.002429 | 0.047619 | 0.047619 | 0.020309 | 0.0046214 |
| K-50 Cosine | | | | | |
| 1 | 0 | 0 | 0 | 0.000888 | 0 |
| 5 | 0.005556 | 0.009524 | 0.009524 | 0.004358 | 0.0070175 |
| 10 | 0.005556 | 0.009524 | 0.009524 | 0.008646 | 0.0070175 |
| 15 | 0.005556 | 0.009524 | 0.009524 | 0.012631 | 0.0070175 |
| 20 | 0.006082 | 0.019048 | 0.019048 | 0.016505 | 0.0092199 |

Tabela 21 - Resultados das avaliações ao método IBCF para a segunda amostra

| Nº recomendações | Precision | Recall | TPR | FPR | F-Measure |
|---------------------|-----------|----------|----------|----------|-----------|
| K-10 Pearson | | | | | |
| 1 | 0.108853 | 0.089706 | 0.089706 | 0.007503 | 0.098356 |
| 5 | 0.042396 | 0.175 | 0.175 | 0.040201 | 0.068256 |
| 10 | 0.031856 | 0.260294 | 0.260294 | 0.078992 | 0.056766 |
| 15 | 0.026908 | 0.307353 | 0.307353 | 0.11127 | 0.049483 |
| 20 | 0.025519 | 0.345588 | 0.345588 | 0.134184 | 0.047529 |
| K-10 Cosine | | | | | |
| 1 | 0.110565 | 0.091176 | 0.091176 | 0.007488 | 0.099939 |
| 5 | 0.046335 | 0.191176 | 0.191176 | 0.040006 | 0.074592 |
| 10 | 0.032367 | 0.264706 | 0.264706 | 0.078571 | 0.057681 |
| 15 | 0.028001 | 0.325 | 0.325 | 0.110864 | 0.05156 |
| 20 | 0.025365 | 0.355882 | 0.355882 | 0.135399 | 0.047354 |
| K-30 Pearson | | | | | |
| 1 | 0.096254 | 0.079412 | 0.079412 | 0.007608 | 0.087025 |
| 5 | 0.044235 | 0.182353 | 0.182353 | 0.040216 | 0.071198 |
| 10 | 0.031344 | 0.258824 | 0.258824 | 0.081092 | 0.055916 |
| 15 | 0.025748 | 0.317647 | 0.317647 | 0.121504 | 0.047634 |
| 20 | 0.022599 | 0.369118 | 0.369118 | 0.161044 | 0.04259 |
| K-30 Cosine | | | | | |
| 1 | 0.101466 | 0.083824 | 0.083824 | 0.007563 | 0.091805 |
| 5 | 0.044594 | 0.183824 | 0.183824 | 0.040171 | 0.071776 |
| 10 | 0.032046 | 0.264706 | 0.264706 | 0.080852 | 0.057171 |
| 15 | 0.026426 | 0.325 | 0.325 | 0.120738 | 0.048877 |
| 20 | 0.023906 | 0.386765 | 0.386765 | 0.159124 | 0.045028 |
| K-50 Pearson | | | | | |
| 1 | 0.099597 | 0.082353 | 0.082353 | 0.007578 | 0.090158 |
| 5 | 0.045938 | 0.189706 | 0.189706 | 0.040156 | 0.073966 |
| 10 | 0.032223 | 0.266176 | 0.266176 | 0.081197 | 0.057486 |
| 15 | 0.026161 | 0.323529 | 0.323529 | 0.122029 | 0.048407 |
| 20 | 0.022852 | 0.375 | 0.375 | 0.16211 | 0.043078 |
| K-50 Cosine | | | | | |
| 1 | 0.097842 | 0.080882 | 0.080882 | 0.007593 | 0.088558 |
| 5 | 0.044575 | 0.183824 | 0.183824 | 0.040171 | 0.071751 |
| 10 | 0.031367 | 0.258824 | 0.258824 | 0.080942 | 0.055953 |
| 15 | 0.02616 | 0.322059 | 0.322059 | 0.120903 | 0.04839 |
| 20 | 0.023101 | 0.375 | 0.375 | 0.159784 | 0.043521 |

Tabela 22 - Resultados das avaliações ao método IBCF para a terceira amostra

Anexo G – Resultados das avaliações para o método UBCF

| Nº recomendações | Precision | Recall | TPR | FPR | F-Measure |
|---------------------|-----------|----------|----------|----------|-----------|
| K-10 Pearson | | | | | |
| 1 | 0.031579 | 0.031579 | 0.031579 | 0.020175 | 0.03 |
| 5 | 0.027368 | 0.136842 | 0.136842 | 0.101316 | 0.05 |
| 10 | 0.022105 | 0.221053 | 0.221053 | 0.203728 | 0.04 |
| 15 | 0.024561 | 0.368421 | 0.368421 | 0.304605 | 0.05 |
| 20 | 0.028132 | 0.557895 | 0.557895 | 0.401754 | 0.05 |
| K-10 Cosine | | | | | |
| 1 | 0.021053 | 0.021053 | 0.021053 | 0.020395 | 0.02 |
| 5 | 0.025263 | 0.126316 | 0.126316 | 0.101535 | 0.04 |
| 10 | 0.022105 | 0.221053 | 0.221053 | 0.203728 | 0.04 |
| 15 | 0.021053 | 0.315789 | 0.315789 | 0.305921 | 0.04 |
| 20 | 0.026842 | 0.536842 | 0.536842 | 0.405482 | 0.05 |
| K-30 Pearson | | | | | |
| 1 | 0.021053 | 0.021053 | 0.021053 | 0.020395 | 0.02 |
| 5 | 0.018947 | 0.094737 | 0.094737 | 0.102193 | 0.03 |
| 10 | 0.015789 | 0.157895 | 0.157895 | 0.205044 | 0.03 |
| 15 | 0.01614 | 0.242105 | 0.242105 | 0.307456 | 0.03 |
| 20 | 0.017368 | 0.347368 | 0.347368 | 0.40943 | 0.03 |
| K-30 Cosine | | | | | |
| 1 | 0.021053 | 0.021053 | 0.021053 | 0.020395 | 0.02 |
| 5 | 0.018947 | 0.094737 | 0.094737 | 0.102193 | 0.03 |
| 10 | 0.015789 | 0.157895 | 0.157895 | 0.205044 | 0.03 |
| 15 | 0.01614 | 0.242105 | 0.242105 | 0.307456 | 0.03 |
| 20 | 0.018421 | 0.368421 | 0.368421 | 0.408991 | 0.04 |
| K-50 Pearson | | | | | |
| 1 | 0.021053 | 0.021053 | 0.021053 | 0.020395 | 0.02 |
| 5 | 0.016842 | 0.084211 | 0.084211 | 0.102412 | 0.03 |
| 10 | 0.014737 | 0.147368 | 0.147368 | 0.205263 | 0.03 |
| 15 | 0.01614 | 0.242105 | 0.242105 | 0.307456 | 0.03 |
| 20 | 0.017895 | 0.357895 | 0.357895 | 0.409211 | 0.03 |
| K-50 Cosine | | | | | |
| 1 | 0.021053 | 0.021053 | 0.021053 | 0.020395 | 0.02 |
| 5 | 0.016842 | 0.084211 | 0.084211 | 0.102412 | 0.03 |
| 10 | 0.014737 | 0.147368 | 0.147368 | 0.205263 | 0.03 |
| 15 | 0.01614 | 0.242105 | 0.242105 | 0.307456 | 0.03 |
| 20 | 0.018421 | 0.368421 | 0.368421 | 0.408991 | 0.04 |

Tabela 23 - Resultados das avaliações ao método UBCF para a primeira amostra

| Nº recomendações | Precision | Recall | TPR | FPR | F-Measure |
|---------------------|-----------|----------|----------|----------|-----------|
| K-10 Pearson | | | | | |
| 1 | 0 | 0 | 0 | 0.001059 | 0.000000 |
| 5 | 0 | 0 | 0 | 0.005297 | 0.000000 |
| 10 | 0 | 0 | 0 | 0.010593 | 0.000000 |
| 15 | 0.000635 | 0.009524 | 0.009524 | 0.01588 | 0.001190 |
| 20 | 0.000476 | 0.009524 | 0.009524 | 0.021176 | 0.000907 |
| K-10 Cosine | | | | | |
| 1 | 0 | 0 | 0 | 0.001049 | 0.000000 |
| 5 | 0 | 0 | 0 | 0.005246 | 0.000000 |
| 10 | 0.001905 | 0.019048 | 0.019048 | 0.010402 | 0.003463 |
| 15 | 0.001937 | 0.028571 | 0.028571 | 0.015476 | 0.003627 |
| 20 | 0.001929 | 0.038095 | 0.038095 | 0.02041 | 0.003671 |
| K-30 Pearson | | | | | |
| 1 | 0 | 0 | 0 | 0.001059 | 0.000000 |
| 5 | 0 | 0 | 0 | 0.005297 | 0.000000 |
| 10 | 0 | 0 | 0 | 0.010593 | 0.000000 |
| 15 | 0 | 0 | 0 | 0.01589 | 0.000000 |
| 20 | 0.000476 | 0.009524 | 0.009524 | 0.021176 | 0.000907 |
| K-30 Cosine | | | | | |
| 1 | 0 | 0 | 0 | 0.001049 | 0.000000 |
| 5 | 0 | 0 | 0 | 0.005246 | 0.000000 |
| 10 | 0.001905 | 0.019048 | 0.019048 | 0.010402 | 0.003463 |
| 15 | 0.001937 | 0.028571 | 0.028571 | 0.015476 | 0.003627 |
| 20 | 0.001452 | 0.028571 | 0.028571 | 0.02042 | 0.002764 |
| K-50 Pearson | | | | | |
| 1 | 0 | 0 | 0 | 0.001059 | 0.000000 |
| 5 | 0 | 0 | 0 | 0.005297 | 0.000000 |
| 10 | 0 | 0 | 0 | 0.010593 | 0.000000 |
| 15 | 0 | 0 | 0 | 0.01589 | 0.000000 |
| 20 | 0 | 0 | 0 | 0.021186 | 0.000000 |
| K-50 Cosine | | | | | |
| 1 | 0 | 0 | 0 | 0.001049 | 0.000000 |
| 5 | 0 | 0 | 0 | 0.005246 | 0.000000 |
| 10 | 0.001905 | 0.019048 | 0.019048 | 0.010402 | 0.003463 |
| 15 | 0.001937 | 0.028571 | 0.028571 | 0.015476 | 0.003627 |
| 20 | 0.001452 | 0.028571 | 0.028571 | 0.02042 | 0.002764 |

Tabela 24 - Resultados das avaliações ao método UBCF para a segunda amostra

| Nº recomendações | Precision | Recall | TPR | FPR | F-Measure |
|---------------------|-----------|----------|----------|----------|-----------|
| K-10 Pearson | | | | | |
| 1 | 0.008824 | 0.008824 | 0.008824 | 0.010114 | 0.008824 |
| 5 | 0.021765 | 0.105882 | 0.105882 | 0.04913 | 0.036107 |
| 10 | 0.026454 | 0.222059 | 0.222059 | 0.087725 | 0.047275 |
| 15 | 0.026322 | 0.254412 | 0.254412 | 0.108748 | 0.047708 |
| 20 | 0.026143 | 0.264706 | 0.264706 | 0.121369 | 0.047586 |
| K-10 Cosine | | | | | |
| 1 | 0.008824 | 0.008824 | 0.008824 | 0.010114 | 0.008824 |
| 5 | 0.022353 | 0.108824 | 0.108824 | 0.04916 | 0.037088 |
| 10 | 0.026617 | 0.223529 | 0.223529 | 0.087455 | 0.04757 |
| 15 | 0.026411 | 0.254412 | 0.254412 | 0.107128 | 0.047854 |
| 20 | 0.02618 | 0.261765 | 0.261765 | 0.118232 | 0.047599 |
| K-30 Pearson | | | | | |
| 1 | 0.013235 | 0.013235 | 0.013235 | 0.010069 | 0.013235 |
| 5 | 0.011471 | 0.057353 | 0.057353 | 0.050435 | 0.019118 |
| 10 | 0.012353 | 0.123529 | 0.123529 | 0.10078 | 0.02246 |
| 15 | 0.014118 | 0.211765 | 0.211765 | 0.1509 | 0.026471 |
| 20 | 0.015086 | 0.301471 | 0.301471 | 0.20096 | 0.028733 |
| K-30 Cosine | | | | | |
| 1 | 0.013235 | 0.013235 | 0.013235 | 0.010069 | 0.013235 |
| 5 | 0.012353 | 0.061765 | 0.061765 | 0.050345 | 0.020588 |
| 10 | 0.013105 | 0.130882 | 0.130882 | 0.100105 | 0.023824 |
| 15 | 0.015999 | 0.238235 | 0.238235 | 0.148544 | 0.029984 |
| 20 | 0.016578 | 0.325 | 0.325 | 0.196023 | 0.031546 |
| K-50 Pearson | | | | | |
| 1 | 0.010294 | 0.010294 | 0.010294 | 0.010099 | 0.010294 |
| 5 | 0.007941 | 0.039706 | 0.039706 | 0.050615 | 0.013235 |
| 10 | 0.008971 | 0.089706 | 0.089706 | 0.101125 | 0.01631 |
| 15 | 0.009706 | 0.145588 | 0.145588 | 0.151576 | 0.018199 |
| 20 | 0.01125 | 0.225 | 0.225 | 0.201786 | 0.021429 |
| K-50 Cosine | | | | | |
| 1 | 0.011765 | 0.011765 | 0.011765 | 0.010084 | 0.011765 |
| 5 | 0.01 | 0.05 | 0.05 | 0.050465 | 0.016667 |
| 10 | 0.011193 | 0.111765 | 0.111765 | 0.1003 | 0.020348 |
| 15 | 0.012371 | 0.183824 | 0.183824 | 0.1491 | 0.023183 |
| 20 | 0.013996 | 0.273529 | 0.273529 | 0.196579 | 0.026629 |

Tabela 25 - Resultados das avaliações ao método UBCF para a terceira amostra

Anexo H – Resultados das avaliações entre todos os métodos

| Nº recomendações | Precision | Recall | TPR | FPR | F-Measure |
|-----------------------------|-----------|----------|----------|----------|-----------|
| AR 2 20 | | | | | |
| 1 | 0.021053 | 0.021053 | 0.021053 | 0.020395 | 0.02 |
| 5 | 0.035789 | 0.178947 | 0.178947 | 0.100439 | 0.06 |
| 10 | 0.033684 | 0.336842 | 0.336842 | 0.201316 | 0.06 |
| 15 | 0.032982 | 0.494737 | 0.494737 | 0.302193 | 0.06 |
| 20 | 0.033684 | 0.673684 | 0.673684 | 0.402632 | 0.06 |
| Itens aleatórios | | | | | |
| 1 | 0.042105 | 0.042105 | 0.042105 | 0.019956 | 0.04 |
| 5 | 0.031579 | 0.157895 | 0.157895 | 0.100877 | 0.05 |
| 10 | 0.032632 | 0.326316 | 0.326316 | 0.201535 | 0.06 |
| 15 | 0.025263 | 0.378947 | 0.378947 | 0.304605 | 0.05 |
| 20 | 0.026842 | 0.536842 | 0.536842 | 0.405482 | 0.05 |
| Itens mais populares | | | | | |
| 1 | 0.052632 | 0.052632 | 0.052632 | 0.019737 | 0.05 |
| 5 | 0.033684 | 0.168421 | 0.168421 | 0.100658 | 0.06 |
| 10 | 0.035789 | 0.357895 | 0.357895 | 0.200877 | 0.07 |
| 15 | 0.034386 | 0.515789 | 0.515789 | 0.301754 | 0.06 |
| 20 | 0.034737 | 0.694737 | 0.694737 | 0.402193 | 0.07 |
| IBCF K-30 Cosine | | | | | |
| 1 | 0.031579 | 0.031579 | 0.031579 | 0.020175 | 0.03 |
| 5 | 0.027368 | 0.136842 | 0.136842 | 0.101316 | 0.05 |
| 10 | 0.022105 | 0.221053 | 0.221053 | 0.203728 | 0.04 |
| 15 | 0.024561 | 0.368421 | 0.368421 | 0.304605 | 0.05 |
| 20 | 0.028132 | 0.557895 | 0.557895 | 0.401754 | 0.05 |
| UBCF K-10 Pearson | | | | | |
| 1 | 0.052632 | 0.052632 | 0.052632 | 0.019737 | 0.05 |
| 5 | 0.042105 | 0.210526 | 0.210526 | 0.099781 | 0.07 |
| 10 | 0.04 | 0.4 | 0.4 | 0.2 | 0.07 |
| 15 | 0.036491 | 0.547368 | 0.547368 | 0.301096 | 0.07 |
| 20 | 0.033158 | 0.663158 | 0.663158 | 0.402851 | 0.06 |

Tabela 26 - Resultados das avaliações a todos os métodos para a primeira amostra

| Nº recomendações | Precision | Recall | TPR | FPR | F-Measure |
|-----------------------------|-----------|----------|----------|----------|-----------|
| AR 2 20 | | | | | |
| 1 | 0.031765 | 0.019048 | 0.019048 | 0.000706 | 0.023815 |
| 5 | 0.012186 | 0.038095 | 0.038095 | 0.002653 | 0.018466 |
| 10 | 0.014186 | 0.057143 | 0.057143 | 0.004217 | 0.02273 |
| 15 | 0.013942 | 0.066667 | 0.066667 | 0.005206 | 0.023061 |
| 20 | 0.013806 | 0.066667 | 0.066667 | 0.00571 | 0.022875 |
| Itens aleatórios | | | | | |
| 1 | 0 | 0 | 0 | 0.001059 | 0 |
| 5 | 0.001905 | 0.009524 | 0.009524 | 0.005287 | 0.003175 |
| 10 | 0.000952 | 0.009524 | 0.009524 | 0.010583 | 0.001732 |
| 15 | 0.000635 | 0.009524 | 0.009524 | 0.01588 | 0.00119 |
| 20 | 0.000952 | 0.019048 | 0.019048 | 0.021166 | 0.001814 |
| Itens mais populares | | | | | |
| 1 | 0 | 0 | 0 | 0.001059 | 0 |
| 5 | 0.005714 | 0.028571 | 0.028571 | 0.005266 | 0.009524 |
| 10 | 0.004762 | 0.047619 | 0.047619 | 0.010543 | 0.008658 |
| 15 | 0.003175 | 0.047619 | 0.047619 | 0.015839 | 0.005952 |
| 20 | 0.002381 | 0.047619 | 0.047619 | 0.021136 | 0.004535 |
| IBCF K-30 Cosine | | | | | |
| 1 | 0.010526 | 0.009524 | 0.009524 | 0.000989 | 0.01 |
| 5 | 0.004737 | 0.019048 | 0.019048 | 0.004863 | 0.007587 |
| 10 | 0.004737 | 0.019048 | 0.019048 | 0.009463 | 0.007587 |
| 15 | 0.006673 | 0.047619 | 0.047619 | 0.0136 | 0.011706 |
| 20 | 0.00627 | 0.047619 | 0.047619 | 0.017464 | 0.011081 |
| UBCF K-10 Pearson | | | | | |
| 1 | 0 | 0 | 0 | 0.001049 | 0 |
| 5 | 0 | 0 | 0 | 0.005246 | 0 |
| 10 | 0.001905 | 0.019048 | 0.019048 | 0.010402 | 0.003463 |
| 15 | 0.001937 | 0.028571 | 0.028571 | 0.015476 | 0.003627 |
| 20 | 0.001929 | 0.038095 | 0.038095 | 0.02041 | 0.003671 |

Tabela 27 - Resultados das avaliações a todos os métodos para a segunda amostra

| Nº recomendações | Precision | Recall | TPR | FPR | F-Measure |
|-----------------------------|-----------|----------|----------|----------|-----------|
| AR 2 20 | | | | | |
| 1 | 0.146522 | 0.067647 | 0.067647 | 0.004067 | 0.09256 |
| 5 | 0.136707 | 0.094118 | 0.094118 | 0.007593 | 0.111483 |
| 10 | 0.136487 | 0.094118 | 0.094118 | 0.007683 | 0.11141 |
| 15 | 0.136487 | 0.094118 | 0.094118 | 0.007683 | 0.11141 |
| 20 | 0.136487 | 0.094118 | 0.094118 | 0.007683 | 0.11141 |
| Itens aleatórios | | | | | |
| 1 | 0.014706 | 0.014706 | 0.014706 | 0.010054 | 0.014706 |
| 5 | 0.011176 | 0.055882 | 0.055882 | 0.05045 | 0.018627 |
| 10 | 0.011618 | 0.116176 | 0.116176 | 0.100855 | 0.021123 |
| 15 | 0.011961 | 0.179412 | 0.179412 | 0.15123 | 0.022426 |
| 20 | 0.011397 | 0.227941 | 0.227941 | 0.201756 | 0.021709 |
| Itens mais populares | | | | | |
| 1 | 0.058824 | 0.058824 | 0.058824 | 0.009604 | 0.058824 |
| 5 | 0.03 | 0.15 | 0.15 | 0.04949 | 0.05 |
| 10 | 0.022941 | 0.229412 | 0.229412 | 0.0997 | 0.041711 |
| 15 | 0.02098 | 0.314706 | 0.314706 | 0.14985 | 0.039338 |
| 20 | 0.018235 | 0.364706 | 0.364706 | 0.20036 | 0.034734 |
| IBCF K-30 Cosine | | | | | |
| 1 | 0.110565 | 0.091176 | 0.091176 | 0.007488 | 0.099939 |
| 5 | 0.046335 | 0.191176 | 0.191176 | 0.040006 | 0.074592 |
| 10 | 0.032367 | 0.264706 | 0.264706 | 0.078571 | 0.057681 |
| 15 | 0.028001 | 0.325 | 0.325 | 0.110864 | 0.05156 |
| 20 | 0.025365 | 0.355882 | 0.355882 | 0.135399 | 0.047354 |
| UBCF K-10 Pearson | | | | | |
| 1 | 0.008824 | 0.008824 | 0.008824 | 0.010114 | 0.008824 |
| 5 | 0.022353 | 0.108824 | 0.108824 | 0.04916 | 0.037088 |
| 10 | 0.026617 | 0.223529 | 0.223529 | 0.087455 | 0.04757 |
| 15 | 0.026411 | 0.254412 | 0.254412 | 0.107128 | 0.047854 |
| 20 | 0.02618 | 0.261765 | 0.261765 | 0.118232 | 0.047599 |

Tabela 28 - Resultados das avaliações a todos os métodos para a terceira amostra