



Sistema de Recomendação em um Aplicativo Móvel para Intervenção do Consumo de Álcool e Tabaco

Fernanda Nunes Dutra

JUIZ DE FORA
DEZEMBRO, 2018

Sistema de Recomendação em um Aplicativo Móvel para Intervenção do Consumo de Álcool e Tabaco

FERNANDA NUNES DUTRA

Universidade Federal de Juiz de Fora

Instituto de Ciências Exatas

Departamento de Ciência da Computação

Mestrado em Bacharelado em Sistemas de Informação

Orientador: Heder Soares Bernardino

JUIZ DE FORA

DEZEMBRO, 2018

SISTEMA DE RECOMENDAÇÃO EM UM APLICATIVO MÓVEL PARA INTERVENÇÃO DO CONSUMO DE ÁLCOOL E TABACO

Fernanda Nunes Dutra

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO DE CIÊNCIAS EXATAS DA UNIVERSIDADE FEDERAL DE JUIZ DE FORA, COMO PARTE INTEGRANTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM BACHARELADO EM SISTEMAS DE INFORMAÇÃO.

Aprovada por:

Heder Soares Bernardino
Doutor em Modelagem Computacional

Luciana Conceição Dias Campos
Doutora em Engenharia Elétrica

Victor Stroele de Andrade Menezes
Doutor em Engenharia de Sistemas e Computação

JUIZ DE FORA

04 DE DEZEMBRO, 2018

Aos pais e irmão, pelo apoio.

Resumo

Um dos diferenciais dos sistemas de *e-health*, como as intervenções destinadas às mudanças comportamentais, é a geração de conteúdo personalizado. Desta forma, sistemas de recomendação podem ser uma excelente solução para este fim. Neste trabalho foi desenvolvido um sistema de *feedback* complementar ao sistema de recomendação desenvolvido para o sistema *web* Viva Sem Tabaco. Além disso foi desenvolvido um aplicativo baseado no *website*, onde serão disponibilizado o conteúdo sugerido pelo sistema de recomendação. O objetivo é que o aplicativo possa coletar mais informações e dessa forma melhorar as sugestões do sistema de recomendação proposto. Além disso, espera-se que o aplicativo possa diminuir a desistência durante o tratamento.

Palavras-chave: sistema de recomendação; filtragem colaborativa; agrupamento; e-health; intervenção; álcool; tabaco

Abstract

One of the differentials of e-health, as the proposals for changes is a generation of personalized themes. In this way, recommender systems can be an excellent solution for this purpose. This article will be developed a feedback system to complementary the recommender system currently use in the interventation for a cessation of smoking. An approach was developed with the use of grouping algorithms as a way to generate sets of users who share of the same profile. The recommender system serves as an helper system, developed to improve an intervention, where the user can evaluete tips. It is expected that after the implementation of the comments the discussion system. results better, being able to learn what is negative for the user.

Keywords:recommender systems; collaborative filtering; clustering; e-health; intervention; alcohol; tobacco; drinking; smoking

Agradecimentos

Agradeço ao Heder Soares Bernardino pela orientação e paciência. Aos meus pais e irmão que sempre me apoiaram nas minhas escolhas e não mediram esforços para me ajudar. Ao Gabriel por todo amor, carinhos e conselhos. E obrigada a todos os amigos que tornaram essa caminhada mais agradável.

“Tell me something, girl

Are you happy in this modern world?”

Lady Gaga (Shallow)

Conteúdo

Lista de Figuras	8
Lista de Tabelas	9
Lista de Abreviações	10
1 Introdução	11
1.1 Justificativa	12
1.2 Objetivos	12
1.3 Estrutura do Trabalho	13
2 Revisão Bibliográfica	14
2.1 Sistemas de Recomendação	14
2.2 Aprendizado de Máquina e Mineração de Dados	18
2.2.1 Agrupamento	18
3 Viva Sem Tabaco	21
3.1 Aplicativo Desenvolvido	22
3.1.1 Modelos do Sistema	22
3.1.2 Telas	25
3.2 Servidor	28
3.2.1 REST Web Service	28
3.2.2 RESTFul API	28
3.2.3 Requisições	29
4 Sistema de Recomendação Proposto	31
4.1 Agrupamento de Usuário	31
4.1.1 Caracterização dos usuários	31
4.1.2 Métricas de Similaridade	32
4.1.3 Algoritmos de Agrupamento	33
4.1.4 Métrica de Qualidade	35
4.2 Sistema de Recomendação	35
4.2.1 Itens Recomendados	37
4.2.2 Cálculo das Predições e Recomendações	37
4.2.3 Execução do Sistema de Recomendação	38
5 Experimentos	39
5.1 Base de Dados	39
5.1.1 Características dos Usuários	40
5.2 Experimento 1	42
5.3 Experimento 2	43
5.4 Experimento 3	44
6 Conclusões e Trabalhos Futuros	47
Bibliografia	49

Lista de Figuras

3.1	Caso de Uso de Login	23
3.2	Caso de Uso de Inserção de Cigarros	23
3.3	Caso de Uso de Inserção de Cigarros	24
3.4	Caso de Uso de Leitura e Curtida de Dicas	24
3.5	Caso de Uso de Desafio	25
3.6	Tela de Login	26
3.7	Telas Evolução e Tela de Hoje	27
3.8	Telas Desafios e Tela de Dicas	27
5.1	Gráfico de Pizza do sexo dos usuários testadores.	40
5.2	Gráfico de Pizza com o número de usuários fumantes, ex-fumantes e não fumantes.	41
5.3	Gráfico de barra da idade dos usuário testadores.	41
5.4	Gráfico de Pizza das avaliações dos usuários para as dicas indicadas aleatoriamente.	44
5.5	Gráfico de Pizza das avaliações dos usuários para as dicas indicadas pelo SR.	45

Lista de Tabelas

2.1	Resumo das técnicas de filtragem. Extraído de (BURKE, 2002).	17
4.1	Dados considerados na recomendação. Extraído de (ROCHA, 2017).	32
4.2	Dados sobre as recomendações salvas no banco	36
5.1	Resultado do coeficiente silhueta para cada algoritmo e métrica	42

Lista de Abreviações

- FC Filtragem Colaborativa
- SR Sistemas de Recomendação
- KDD *Knowledge Discovery in Databases*

1 Introdução

Segundo Bobadilla et al. (2013) os sistemas de recomendações têm como objetivo inferir preferências e necessidades do usuário e sugerir itens mais adequados de acordo com seu perfil. As recomendações podem ser feitas analisando dados dos usuários e dos itens. O principal desafio é recomendar conteúdo útil ao usuário. Diferentes técnicas e algoritmos podem ser explorados para obter maior qualidade nas recomendações.

Atualmente, os sistemas de recomendação são utilizados para filmes, músicas, livros e comércio eletrônico. Porém, eles também podem ser estendidos a qualquer contexto em que se busque personalização de conteúdo (LU et al., 2015).

O Departamento de Ciência da Computação juntamente com o Centro de Referência em Pesquisa, Intervenção e Avaliação em Álcool e Outras Drogas (CREPEIA) da Universidade de Juiz de Fora desenvolveram as intervenções *on-line* Viva Sem Tabaco ¹ e Álcool e Saúde², que oferecem conteúdo gratuito para ajudar as pessoas interessadas em cessar o consumo de álcool e tabaco. De forma a melhorar a intervenção foi desenvolvido um sistema de recomendação próprio que indica informações úteis aos usuários dos sistemas (ROCHA, 2017).

Apesar do desenvolvimento, o sistema atual não envia as recomendações aos usuários. Propõe-se aqui o aperfeiçoamento do sistema de recomendação já existente, adicionando o envio de recomendações ao usuário. O objetivo é inserir etapas que de irão refinar as recomendações. Como por exemplo, um sistema de *feedback* dos usuários, a fim de observar a qualidade do conteúdo recomendado e utilizar este retorno para aprimorar as recomendação.

Com o sistema de *feedback*, poderemos aprender o que é negativo para cada usuário. Isso evita os falsos positivos, ou seja, itens não relevantes mas que o sistema julga como o contrário.

¹<<http://www.vivasemtabaco.com.br>>

²<<http://www.alcoolesaude.com.br>>

1.1 Justificativa

Sistemas voltados para tratamentos médicos são cada vez mais comuns. As principais vantagens do *e-health* são o fácil acesso ao usuário e a produção personalizada de conteúdo (ROCHA, 2017). Os sistemas de recomendação estão diretamente ligados a esse assunto já que podem ser utilizados para produção de conteúdo específico para cada usuário individualmente.

Um usuário que recebe sugestões de conteúdo de seu interesse pode ficar mais motivado a prosseguir o tratamento. Mesmo que haja uma variedade restrita de conteúdo, é importante que cada pessoa que busca o tratamento receba o conteúdo que melhor se encaixa em seu perfil.

Hoje em dia é utilizada uma abordagem via agrupamento de usuários em sistemas de recomendação para as intervenções Viva Sem Tabaco e Álcool e Saúde. Ambos realizam recomendações de páginas de conteúdo informativo e educativo, mas ainda não enviam o conteúdo ao usuário.

1.2 Objetivos

Este trabalho busca melhorar o sistema de recomendação por meio do uso do sistema de *feedback*. Com o *feedback* do usuário, foi possível informar ao sistema se foi uma boa recomendação ou não, melhorando cada vez mais a informação oferecida ao usuário. Além disso, almeja-se o aperfeiçoamento do tratamento, tornando-o mais interessante e motivador. Como objetivos específicos propõe-se:

- Alterar o sistema de recomendação para recomendar dicas que irão ajudar o usuário durante o tratamento;
- Realizar o envio de recomendações de conteúdo ao usuário;
- Desenvolver o aplicativo do Viva Sem Tabaco e inserir o sistema de *feedback* dentro dele;
- Desenvolver um servidor capaz de auxiliar o aplicativo;

1.3 Estrutura do Trabalho

O conteúdo do trabalho está dividido em quatro capítulos, a seguir cada um deles é introduzido de forma breve.

O capítulo 1 introduz o tema em que o atual trabalho está envolvido. É apresentado ao leitor o problema que foi trabalhado e sua motivação.

O capítulo 2 envolve a revisão bibliográfica dos principais tópicos envolvidos: sistemas de recomendação, aprendizagem de máquina, mineração de dados e agrupamentos. São apresentados artigos importantes presentes na literatura e uma breve definição de conceitos necessários.

O capítulo 3 trata sobre o aplicativo Viva Sem Tabaco, como ele foi desenvolvido e o servidor por trás do aplicativo. Além disso, apresenta-se, também, as telas e os casos de uso base da aplicação.

O capítulo 4 tem como objetivo apresentar como a pesquisa foi conduzida. São definidos os algoritmos, métricas e técnicas usados no trabalho.

O capítulo 5 apresenta os experimentos realizados no sistema, após o desenvolvimento do trabalho.

Por fim, no capítulo 6 apresenta as considerações finais, sintetizando a proposta e seus resultados de maneira geral. São indicados as principais conclusões e propostos trabalhos futuros.

2 Revisão Bibliográfica

2.1 Sistemas de Recomendação

Os Sistemas de Recomendações (SR), em desenvolvimento desde o início dos anos 90, sugerem itens de interesse para as pessoas baseado em seu perfil e em suas preferências (BENNETT; LANNING et al., 2007). Caso alguém solicite uma recomendação de um livro a outras pessoas, ela ficará restrita ao número de pessoas conhecidas. Com o sistema de recomendação é possível ampliar a quantidade informações que serão consultadas antes de realizar uma recomendação. Ou seja, utiliza-se todos os dados existentes durante o processo de recomendação para criar uma recomendação para um usuário. Além disso, os sistemas de recomendação analisam históricos e padrões do usuário que receberá a recomendação, informações que talvez não seriam reconhecidas pelo próprio usuário alvo (EKSTRAND; RIEDL; KONSTAN, 2010).

O Sistema de Recomendação revelou-se ótimo para a indústria do *e-commerce*.

A *Amazon.com*, por exemplo, vem usando há mais de uma década Filtragem Colaborativa (FC) para recomendar seus produtos (EKSTRAND; RIEDL; KONSTAN, 2010). Em 1990, a Filtragem Colaborativa (FC) começou a ser uma solução para lidar com o grande número de dados online. FC ajuda nas escolhas das pessoas baseando-se nas opiniões de outras (RESNICK et al., 1994). Ela permite que os usuários forneçam avaliações de um conjunto de itens (músicas, livros, vídeos), e quando houver dados suficientes no sistema, é possível utilizar as informações extraídas desse usuário para recomendar itens para outro com mesmo perfil (BOBADILLA et al., 2013).

GroupLens, software desenvolvido pelo laboratório de pesquisa *GroupLens*³ do Departamento de Ciência da Computação e Engenharia da Universidade de *Minnesota*, foi um dos primeiros trabalhos na área de SR. Foi utilizado o método de Filtragem Colaborativa para recomendar artigos de interesse de um determinado usuário. O sistema ajuda as pessoas a encontrarem artigos que provavelmente irão gostar. Os clientes leitores classificam os arti-

³<https://grouplens.org/>

gos após lê-los, e o serviço *Better Bit Bureaus* coleta e divulga as classificações. Os serviços partem da heurística de que pessoas que concordaram no passado concordarão novamente. Os usuários podem proteger sua privacidade introduzindo pseudônimos em suas avaliações (RESNICK et al., 1994).

Existem duas variações da Filtragem Colaborativa: Filtragem Colaborativa Baseada em Usuário e Filtragem Colaborativa Baseada em Itens (EKSTRAND; RIEDL; KONSTAN, 2010).

A primeira abordagem, utilizada pelo *GroupLens*, busca usuários que possuem comportamentos semelhantes ao usuário atual e utiliza as avaliações feitas pelo usuário similar para cada item e tenta prever o que o usuário atual poderá gostar (RESNICK et al., 1994). A Filtragem Colaborativa Baseada em Usuário, apesar de eficaz, sofre com o problema de escalabilidade à medida que a base de usuários cresce. Por isso, foi necessário ampliar a FC para grandes bases de usuários, facilitando a implantação em *e-commerce*.

A Filtragem Colaborativa Baseada em Item trata o problema citado anteriormente e é uma das mais utilizadas atualmente (EKSTRAND; RIEDL; KONSTAN, 2010). Ao invés de utilizar a semelhança de comportamento das classificações dos usuários para prever preferências, este método utiliza as semelhanças entre padrões de itens (EKSTRAND; RIEDL; KONSTAN, 2010). Assume-se que um usuário está mais propenso a comprar itens que são semelhantes ou relacionados aos itens que ele já comprou (YAO; CAI, 2017).

Um método diferente dos anteriores é a Filtragem Baseada em Conteúdo, que faz recomendações utilizando atributos dos itens (cor, textura, forma e etc) que um determinado usuário gostou no passado (BALABANOVIĆ; SHOHAM, 1997). O usuário, por sua vez, tem o perfil construído através de questionários, classificações de produtos e histórico de navegação (YAO; CAI, 2017). Diferente da FC, a Filtragem Baseada em Conteúdo seleciona itens com base na correlação entre o conteúdo dos itens e as preferências do usuário e não com base na correlação entre pessoas com preferências semelhantes (METEREN; SOMEREN, 2000).

Ambos os métodos possuem suas limitações. As duas abordagens, Filtragem Colaborativa e Filtragem Baseada em Conteúdo precisam, primeiro, aprender as preferências dos usuários a partir das classificações que eles fazem. Após essa etapa as filtragens poderão

fazer recomendações coerentes.

Como a Filtragem Colaborativa e a Baseada em Conteúdo dependem muito das avaliações dos usuários para fazer as recomendações, um novo item aguarda até ter inúmeras avaliações para que o sistema seja capaz de recomendá-lo (ADOMAVICIUS; TUZHILIN, 2005). Este é um problema comum, conhecido como partida fria (SCHEIN et al., 2002). Existem relativamente poucas informações sobre cada usuário, o que dificulta que o sistema de recomendação seja capaz de gerar resultados específicos para eles (ROBERTS; BROOKS; SHIPWAY, 2014).

Uma forma de lidar com o problema de partida fria é pedir classificações a todos os novos usuários. Solicitando que seja feita uma classificação de itens - previamente escolhidos - pelo usuário. Essa abordagem permite obter informações iniciais sobre o usuário.

Existem outras estratégias que tratam esse problema, uma delas é a estratégia aleatória. O método seleciona itens aleatoriamente. Dessa forma o sistema de recomendação irá aprender após cada avaliação do item recomendado aleatoriamente. Como os itens são indicados de forma aleatória a precisão dos itens indicados é baixa.

Além da partida fria, ambas sofrem outros dois problemas devido aos seus tipos de sistema de recomendações: esparsidade e *ramp-up*. A esparsidade é a ausência de usuários e de classificações de item. Já o problema de *ramp-up* é o fato de que só haverá boas recomendações quando houver um grande número de classificações dos itens a serem recomendados (BURKE, 2002).

O sucesso de uma recomendação por métodos colaborativos depende de uma massa crítica de usuários. Por exemplo, pode haver itens que foram avaliados por poucas pessoas e que, conseqüentemente, seriam recomendados raramente, mesmo que suas avaliações sejam muito boas. Outra questão é o fato de que os usuários que fogem do padrão do resto da população, não terão usuários com perfis semelhantes, o que levará a recomendações fracas (ADOMAVICIUS; TUZHILIN, 2005).

Uma maneira de contornar o problema é utilizar informações do perfil do usuário para calcular as similaridades. Dois usuários não seriam classificados como semelhantes apenas por classificar itens de forma parecida. Também seriam levados em conta dados demográficos como sexo, idade, código de área e educação. Essa extensão da técnica de FC

é conhecida como Filtragem Demográfica (ADOMAVICIUS; TUZHILIN, 2005). Tal método parte do princípio de que indivíduos com atributos pessoais comuns também terão preferências comuns (BOBADILLA et al., 2013).

Um resumo das técnicas mencionadas acima pode ser visto na Tabela 2.1.

Tabela 2.1: Resumo das técnicas de filtragem. Extraído de (BURKE, 2002).

Técnicas	Prós	Contras
Filtragem Colaborativa	A. Identifica gêneros cruzados B. Não é necessário domínio de conhecimento C. Adaptativo: Qualidade melhora com o tempo D. <i>Feedback</i> implícito é suficiente	I. Problema de <i>ramp-up</i> em novo usuários J. Problema de <i>ramp-up</i> em novos itens K. Qualidade dependente de grande conjunto de dados históricos L. Estabilidade
Filtragem Baseada em Conteúdo	B,C,D	I,K,L
Filtragem Demográfica	A,B,C	I,K,L M. Deve reunir informações demográficas

Para superar a fraqueza de cada método, vem sendo utilizada a Abordagem Híbrida que combina duas ou mais técnicas, normalmente de Filtragem Baseada em Conteúdo e Filtragem Colaborativa (ADOMAVICIUS; TUZHILIN, 2005). O Sistema Híbrido pode incorporar as vantagens de ambas e tentar aliviar as desvantagens (BALABANOVIĆ; SHOHAM, 1997).

Dentre tantas opções de técnicas e métodos pode ficar difícil escolher a melhor. É importante salientar que a escolha deve ser fortemente dependente do contexto da aplicação, pois cada situação será mais adequada a uma determinada técnica (LU et al., 2015).

Devido à falta de informações na base de dados sobre as preferências dos usuários, é necessário tratar o problema de partida fria ao recomendar uma dica. Nesse trabalho será utilizada uma estratégia de filtragem colaborativa, com uma recomendação randômica quando houver falta de dados sobre o usuários e usuários similares.

No Capítulo 4 serão explicadas com mais detalhes as abordagens utilizadas no sistema de recomendação.

2.2 Aprendizado de Máquina e Mineração de Dados

O Aprendizado de Máquina estuda métodos capazes de adquirir conhecimento de forma automática (MITCHELL, 1999). O objetivo é fornecer níveis crescentes de automação no processo da Engenharia de Conhecimento de forma a substituir a atividade humana, muito demorada, por técnicas automáticas que aumentam a eficácia e precisão (LANGLEY; SIMON, 1995), utilizando programas de computador que melhoram seu desempenho através da experiência (MITCHELL, 1999). Os processos de aprendizagem podem ser classificados em dois grupos: aprendizado supervisionado e não supervisionado (SATHYA; ABRAHAM, 2013).

O aprendizado supervisionado baseia-se em uma amostra de dados, na qual a classificação correta já é conhecida (SATHYA; ABRAHAM, 2013). Exigem como entrada um *corpus* (documento) de treino com exemplos corretamente identificados que se deseja aprender a mapear (BRAGA; CARVALHO; LUDERMIR, 2000). O objetivo é que o modelo gerado seja capaz de produzir saídas corretas para novas entradas não apresentadas antes (PELLUCCI et al., 2011).

Por outro lado, o aprendizado não supervisionado não utiliza referências (PELLUCCI et al., 2011). Nesse aprendizado apenas os atributos de entrada estão disponíveis (SOUTO et al., 2003). Dessa forma, o algoritmo de aprendizado de máquina fica incumbido de tentar agrupar os dados de acordo com suas características da melhor maneira possível (conhecido como agrupamento) (PARDO; NUNES, 2002).

Algoritmos de aprendizado aplicáveis à minerações de dados incluem redes neurais, programação lógica indutiva e métodos interpretáveis. Embora esses algoritmos sejam relevantes no processo de mineração de dados, é importante ressaltar que o processo envolve outras etapas importantes, como a construção e manutenção do banco de dados, formatação e limpeza de dados (MITCHELL, 1999).

2.2.1 Agrupamento

Nos últimos 30 anos, as técnicas para agrupamento têm sido aplicadas em diversas áreas, como medicina, química, estudos sociais e outras (NG; HAN, 1994). Agrupamento é um método de aprendizado não supervisionado com o objetivo de separar um conjunto finito

de dados não rotulados. Os algoritmos de agrupamento separam os dados em certo número de subconjuntos considerando a semelhança entre si, mantendo dados similares em um mesmo grupo e dados diferentes em outros grupos. As similaridades são calculadas de acordo com uma métrica baseada nas características das variáveis dos objetos (XU; WUNSCH, 2005).

Agrupamento pode ser dividido em duas grandes categoriais, os métodos hierárquicos e os métodos particionais (NG; HAN, 1994).

Os métodos hierárquicos são técnicas onde os dados são particionados sucessivamente (NG; HAN, 1994). A técnica interliga amostras por suas associações, produzindo um dendrograma onde as amostras semelhantes, segundo as variáveis escolhidas, são agrupadas entre si. A suposição básica de sua interpretação é esta: quanto menor a distância entre os pontos, maior a semelhança entre as amostras (NETO; MOITA, 1998).

Existem duas versões dos métodos hierárquicos, os aglomerativos e os divisivos. Nos métodos aglomerativos os objetos estão inicialmente isolados e são progressivamente reunidos em grupos sucessivos até formarem um único grupo. Diferente do anterior, os métodos divisivos iniciam com um único grupo que é dividido em subgrupos até alcançar objetos isolados (LINDEN, 2009).

Nos métodos particionais os padrões são agrupados em um número k de grupos pré-definidos (XU; WUNSCH, 2005). O fato do número de agrupamentos serem escolhidos anteriormente poderá levar a interpretações erradas sobre a estrutura de dados caso o número de agrupamentos não seja ideal. Uma das vantagens desses métodos é a possibilidade de um padrão poder mudar de agrupamento com a evolução do algoritmo (FUNG, 2001). Um exemplo de métodos particionais é o K-means. O K-means e suas variantes têm complexidade de tempo linear, $O(n)$ e é utilizado por causa de sua eficiência em tempo de execução.

Em sistemas de recomendação, os algoritmos de agrupamentos são utilizados para melhorar a qualidade das recomendações. Podem ser feitos agrupamentos de usuários, agrupamentos de itens ou um método híbrido entre os dois agrupamentos anteriores. A aplicação de técnicas de agrupamento reduz a dispersão e melhora a escalabilidade dos sistemas, uma vez que a similaridade pode ser calculada apenas para os usuários nos mesmos

grupos. O agrupamento de usuários e itens também é uma boa alternativa para contornar problemas como a esparsidade dos dados e a partida fria (ZHU; GONG, 2009).

3 Viva Sem Tabaco

O Viva Sem Tabaco é um sistema *web* para intervenção, desenvolvido para auxiliar fumantes no processo de parada, através de uma intervenção para o tratamento de tabagismo.

O sistema foi desenvolvido para ser usado em uma única sessão. Ao acessar a intervenção e criar uma conta, o fumante pode escolher entre conteúdos organizados de acordo com os estágios do modelo transteórico de mudança de comportamentos (Instituto Nacional do Câncer, 2001; U.S Department of Health Services, 2008): "vale a pena parar?", "pronto para parar?" e "Já parei de fumar". A intervenção foi desenvolvida para ser usada em uma sessão. O acompanhamento do fumante é feito através de mensagens de e-mail. Diferentes técnicas de mudança de comportamento foram usadas tais como informações sobre consequências à saúde, foco em sucessos anteriores, definição de objetivos, apoio farmacológico e resolução de problemas. Maiores detalhes sobre a intervenção e seu desenvolvimento podem ser consultados em Gomide et al. (2016).

No entanto, o sistema *web* tem pouca interação com os usuários, gerando baixo engajamento dos usuários e desistência do tratamento. Ademais, com a popularização do uso dos aplicativos, uma forma de aumentar a interação entre o usuário e o Viva Sem Tabaco é desenvolvendo uma aplicação *mobile*. Dessa forma, é possível coletar mais dados do usuário e desenvolver tratamentos cada vez mais adequados.

Com esse objetivo, foi desenvolvido o protótipo de um aplicativo do Viva Sem Tabaco, baseado no *website* de mesmo nome. O protótipo focou na *gameficação* do processo, buscando maior interação com o usuário (PEREIRA, 2017). Apesar do protótipo aplicar elementos de *gameficação*, a nova versão do aplicativo, desenvolvida nesse trabalho não aplicou estes elementos.

O sistema busca ajudar pessoa no tratamento de tabagismo através e dicas enviadas de acordo com o perfil de cada usuário. Apesar de ainda não possuir um sistema de *gameficação*, o aplicativo já envia desafios para o usuário, não realizando a pontuação.

3.1 Aplicativo Desenvolvido

A fim de implementar o envio das recomendações, foi desenvolvido neste trabalho uma nova versão do aplicativo Viva Sem Tabaco. Ao contrário do aplicativo ⁴ desenvolvido no projeto (PEREIRA, 2017), a versão aqui apresentada está disponível na *playstore* ⁵ para os usuários do site utilizarem.

A fim de realizar a sincronização dos dados do aplicativo com o servidor há um serviço sendo executado em *background* que começa a rodar sempre que o usuário acessa o aplicativo. O serviço *background* é responsável por baixar os desafios disponíveis no servidor e que ainda não foram baixados; por enviar os dados diários de cigarros que o usuário insere; e por baixar as dicas, é baixada uma dica por dia e não pode ser repetida.

A estratégia de baixar uma dica por dia foi escolhida para que o sistema de recomendação possua mais dados sobre os usuários na próxima recomendação; já que o usuário poderá ler a dica diária informar se gostou ou não e esses dados serem enviado ao servidor. No próximo dia o sistema de recomendação terá mais dados desse usuário e de outros usuários com perfil similar, aumentando as chances de uma recomendação assertiva.

3.1.1 Modelos do Sistema

Existem três atores principais no sistema, o Usuário Fumante que acessa o aplicativo em busca de tratamento, o próprio Aplicativo que realiza os cálculos e os envios dos dados para o servidor e o Servidor, responsável por persistir os dados e enviar dados que serão utilizados no aplicativo.

Para utilizar o sistema é necessário fazer o *login* no aplicativo, e só poderá fazer *login* no aplicativo os usuários que possuem conta no site Viva Sem Tabaco. É necessário informar o *e-mail* e senha. Ao realizar a ação de *login*, o sistema envia uma requisição ao servidor com as informações digitadas na tela. O servidor precisa autenticar o acesso. Caso tenha sucesso ao autenticar, o usuário poderá ter acesso ao sistema (Figura 3.1).

⁴Código disponível em <<https://github.com/FernandaNunesDutra/livewithouttobaccoapp>>

⁵Aplicativo disponível para teste em <<https://play.google.com/apps/internaltest/4700695285609175039>>

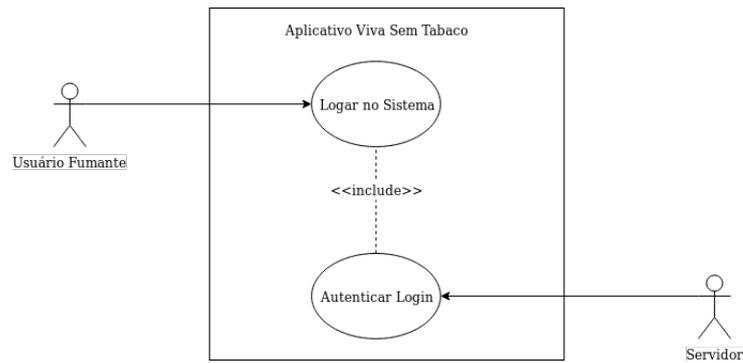


Figura 3.1: Caso de Uso de Login

O usuário insere o valor do maço e quantos cigarros ele fumou no dia corrente. O aplicativo é responsável por realizar os cálculos de gastos e economizados a partir dessas informações. Além disso, o aplicativo deve enviar esses dados ao servidor. O servidor, por sua vez, deve persistir as informações recebidas (Figura 3.2).

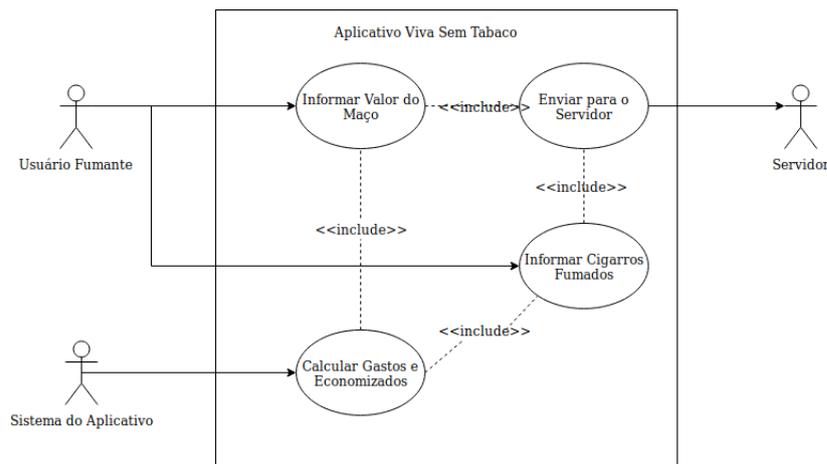


Figura 3.2: Caso de Uso de Inserção de Cigarros

Uma parte importante do aplicativo é poder visualizar toda sua trajetória desde o início do uso do aplicativo. Para que o usuário possa verificar tudo que ele já gastou, economizou e deixou de fumar até o dia de hoje. As informações dos totais são requisitados pelo aplicativo ao servidor, para que ele faça o processamento de todos os dados e retorne a informação para o aplicativo (Figura 3.3).

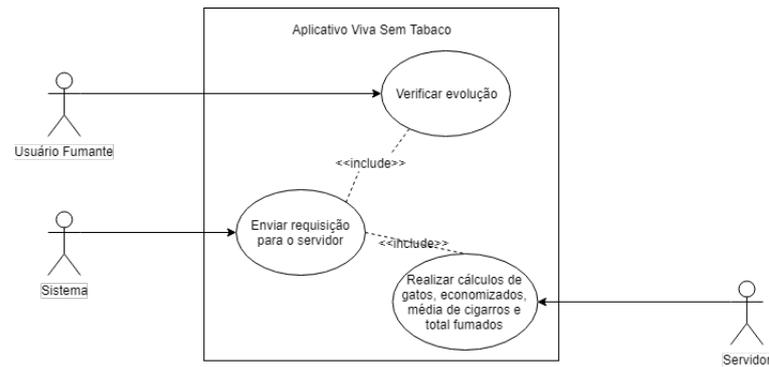


Figura 3.3: Caso de Uso de Inserção de Cigarros

O aplicativo realiza requisições em *background* para obter a dica recomendada no dia corrente. O usuário poderá ler a dica e informar se gostou ou não dela.

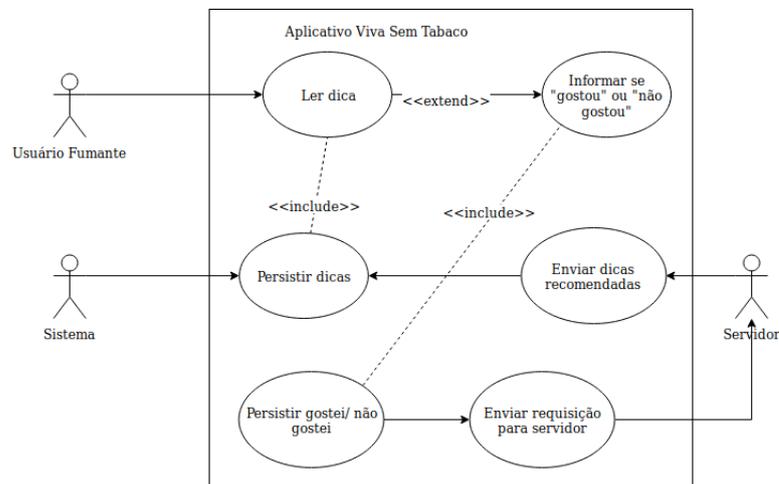


Figura 3.4: Caso de Uso de Leitura e Curtida de Dicas

O caso de uso de desafios (Figura 3.5) é muito próximo ao da dica (Figura 3.4). O usuário lê o desafio, deve realizá-lo e o sistema persistí-lo. Atualmente, o aplicativo não está fazendo a contagem de pontos e a verificação da realização do desafio.

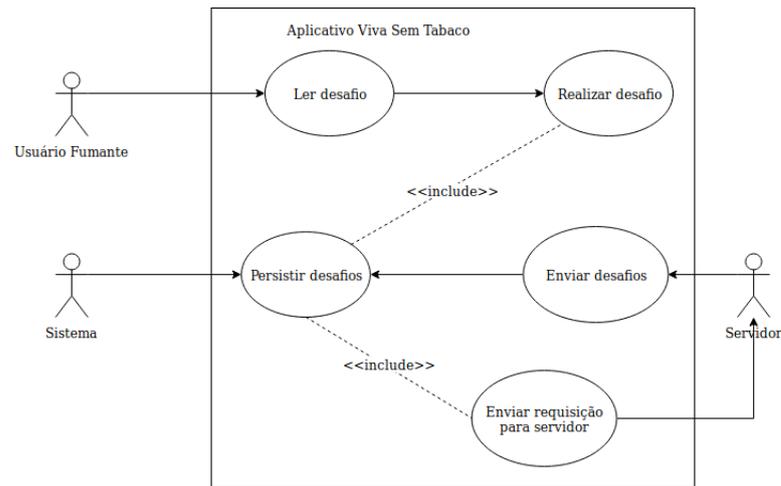


Figura 3.5: Caso de Uso de Desafio

Na Seção 3.1.2 serão mostradas as telas implementadas para cada caso de uso apresentado nessa seção.

3.1.2 Telas

As telas do aplicativo foram desenvolvidas com a intenção de tornar o uso amigável e intuitivo. Foram desenvolvidas cinco telas principais:

- Tela de login (Figura 3.6) é a tela que o usuário do site Viva Sem Tabaco poderá acessar o aplicativo. A tela de login possui *layout* muito próximo do site, para utilizarmos a mesma identidade visual do sistema *web*, utilizou-se como imagem de fundo do aplicativo a imagem existente na tela inicial do *site*;
- Tela de desafios (Figura 3.8), onde o usuário pode ver os desafios disponíveis e quais ele já conquistou;
- Tela de dicas (Figura 3.8), com a listagem das dicas com a opção de informar se gostou ou não da dica apresentada. Nessa tela são coletados os dados de *feedback* do usuário através da seleção da imagem que marcam que a dica foi positiva ou negativa;
- Tela de evolução (Figura 3.7), onde o usuário poderá verificar seus valores totais e sua evolução desde do início da utilização do aplicativo. Como próprio nome diz, essa tela foi construída com o intuito de mostrar ao usuário seu progresso e a evolução durante

o tratamento, mostrando os totais economizados (cálculo do valor economizado é em cima de quantos cigarros ele deixou de fumar baseando na sua própria média), totais gastos e totais fumados. O gráfico existente na página mostra a média de cigarros fumados pelo usuário e a média geral;

- Por fim, a tela de hoje (Figura 3.7), onde o usuário informa o quanto foi consumido no dia atual e o valor do maço de cigarro que ele consome.



Figura 3.6: Tela de Login

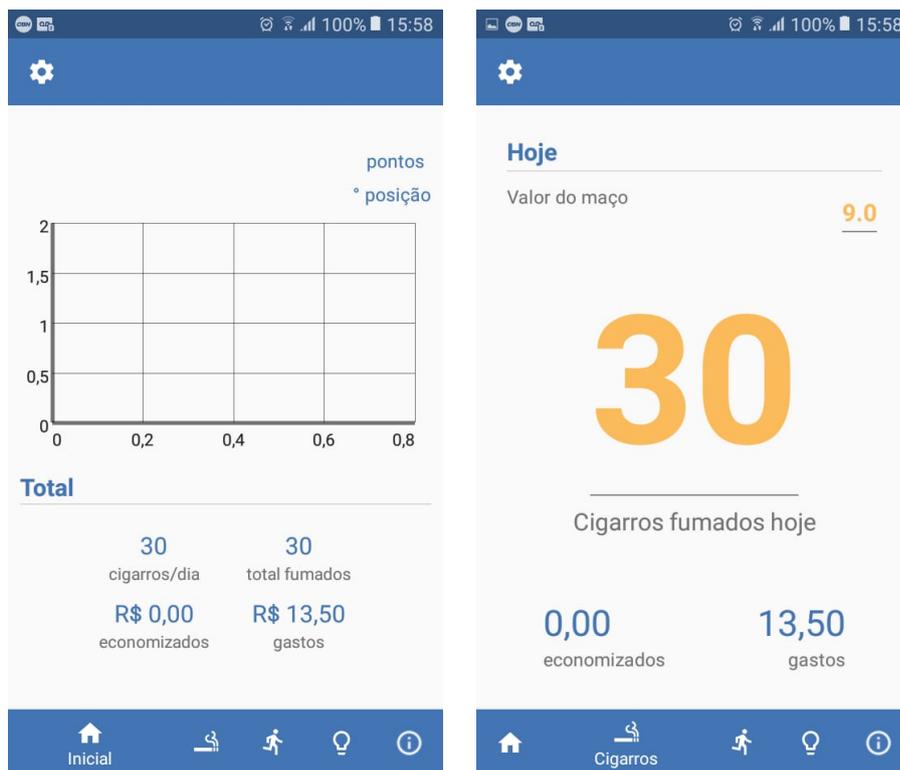


Figura 3.7: Telas Evolução e Tela de Hoje

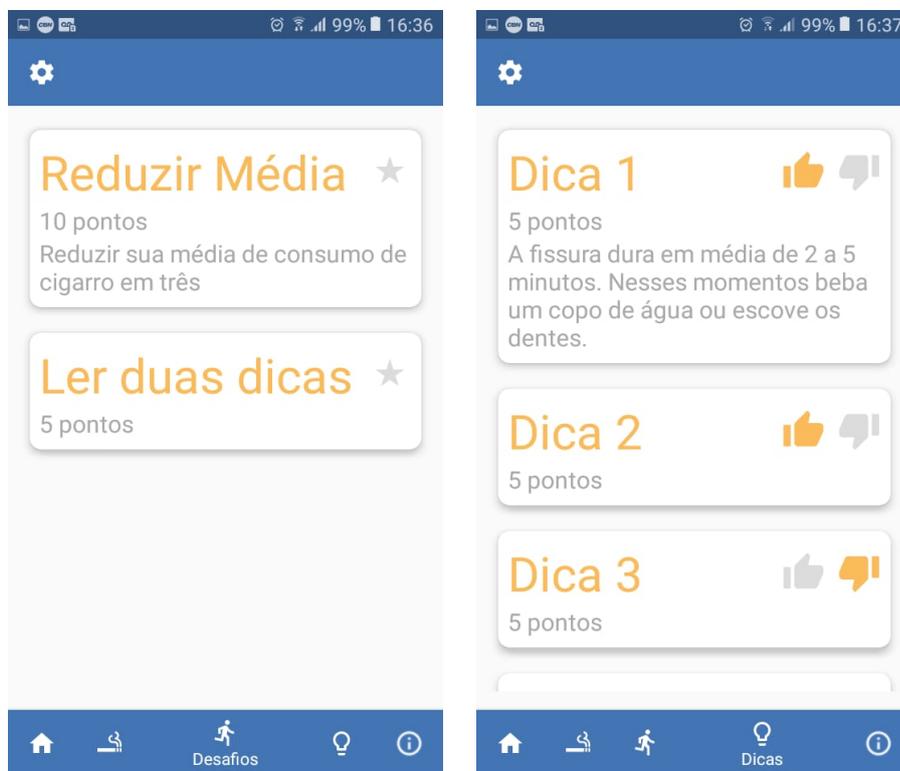


Figura 3.8: Telas Desafios e Tela de Dicas

3.2 Servidor

O servidor⁶ é responsável por armazenar os dados dos usuários e fornecer ao aplicativo as informações necessárias para que ele possa interagir com o usuário. Como por exemplo, as dicas, dados sobre evolução, entre outras. Nele são consistidas as informações diárias do consumo de cigarros. O sistema também é responsável por fornecer os dados de dicas e desafios, além de realizar os cálculos dos totais consumidos e gastos.

3.2.1 REST Web Service

Web Services surgiram com a tendência da arquitetura de software de desenvolver componentes independentes de plataforma. Eles conectam computadores e dispositivos, utilizando a Internet para realizar troca de dados (FENSEL; BUSSLER, 2002).

REST é um estilo arquitetural aplicado ao desenvolvimento de *web service*. O REST se concentra nos papéis dos componentes, ignorando detalhes de sintaxe de protocolo. Nessa arquitetura os dados são enviados através de uma interface HTTP. O cliente é capaz de acessar o recurso através de uma URL exclusiva e um recurso é retornado. O REST é em geral mais rápido, mais leve e pode ser implementado usando diversas ferramentas. No entanto, o cliente precisa saber o que enviar e o que irá receber (PAI, 2014).

3.2.2 RESTFul API

RESTFul API são um conjunto de princípios que definem regras arquiteturais para serviços. Uma das características principais é o uso dos métodos HTTP (RODRIGUEZ, 2008):

- GET é utilizado por um cliente para recuperar informações, através de uma consulta a ser executada no servidor;
- POST é utilizado para criar um novo objeto, nesse caso deve-se enviar no corpo da requisição o novo objeto;
- PUT é utilizado para alterar um objeto.

⁶Código disponível em <<https://github.com/FernandaNunesDutra/wati-api-spring>>

Para ser capaz de acessar os serviços disponíveis na API é necessário fazer uma requisição a partir de um *endpoint*. O *endpoint* é uma url com o caminho do servidor em que o serviço está disponibilizado mais o caminho até o método requisitado. Exemplos dos *endpoints* utilizados na aplicação serão apresentados na Seção 3.2.3.

Na aplicação foi utilizado o formato JSON para envio e retorno dos objetos. Ao fazer uma requisição POST, o cliente envia no corpo da requisição um objeto JSON a ser lido pelo servidor. Além disso, todas as requisições, exceto o login, exigem um *token*. Cada usuário recebe um *token* após realizar um login com sucesso. Esse *token* deve ser enviado através do cabeçalho de todas as requisições pós-login. O *token* serve para validação do usuário e para garantir a segurança das requisições.

3.2.3 Requisições

Os *endpoints* criados para serem chamados no aplicativo estão descritos nessa seção.

Login

Esta requisição é do tipo POST, necessita de um corpo, onde devem conter as informações de *e-mail* e senha do usuário. Ele retorna um *token* que será utilizado em todas as próximas requisições. O *endpoint* dessa requisição é `watiapi/api/user/login`.

Logout

Sair do aplicativo. Requisição do tipo GET, com o *endpoint* `watiapi/api/user/logout`.

Desafios

Retornar a lista de desafios inseridos após a data desejada. Requisição do tipo GET, com o *endpoint* `/watiapi/api/challenge/all?date=[date]`.

Dicas

Retornar a dica recomendada pelo sistema na data desejada. O servidor obtém a dica através de uma resposta do sistema de recomendação. Requisição do tipo GET, com o *endpoint* `/watiapi/api/tip/all?date=[date]`.

Receber e persistir as dicas que receberam “gostei” ou “não gostei” do usuário. Essa requisição é do tipo POST, no corpo é necessário o identificador da dica, o identificador do usuário e o campo informando se gostou ou não da dica. O *endpoint* é `watiapi/api/tipuser/-like`.

Cigarros

Retornar todos os dados sobre cigarros fumados, valor do maço, economizado e gastos do dia. É um método do tipo GET, com *endpoint* `/watiapi/api/cigarette/today`.

Inserir todos os dados sobre cigarros fumados, valor do maço, economizado e gastos do dia. No corpo é necessário o identificador do usuário e os valores a serem inseridos. É um método do tipo POST, com *endpoint* `/watiapi/api/cigarette/today`.

Retornar todos os dados sobre número de cigarros fumados desde que começou a utilizar o aplicativo, média de cigarros fumados por dia, totais gastos e economizados. É um método do tipo GET, com *endpoint* `/watiapi/api/cigarette/total`.

4 Sistema de Recomendação Proposto

4.1 Agrupamento de Usuário

Em um sistema de recomendação com uma abordagem colaborativa baseada em usuário, a prioridade dos itens a serem recomendados aos usuário é calculada a partir de elementos coletados de usuários similares e que podem ter opiniões parecidas (EKSTRAND; RIEDL; KONSTAN, 2010).

Uma alternativa para identificar os usuários similares ao usuário que receberá a recomendação é através de técnicas de agrupamento. Dessa forma pode-se gerar grupos para os usuários do sistema e recomendar uma dica que um outro usuário do mesmo grupo gostou.

Os algoritmos de agrupamento utilizados atualmente no sistema são o *K-means* e o hierárquico aglomerativo (ROCHA, 2017).

4.1.1 Caracterização dos usuários

Espera-se que a partir das características dos usuários, os algoritmos de agrupamento unam usuários com comportamento e perfis similares. Os atributos levados em consideração na base do Viva Sem Tabaco estão disponível na Tabela 4.1.

Tabela 4.1: Dados considerados na recomendação. Extraído de (ROCHA, 2017).

Campos	Tipo	Valor
Idade	Inteiro	{18, . . . , 100}
Sexo	Booleano	{0, 1}
Motivação para parar de fumar	Inteiro	{1, . . . , 10}
Se escolheu como estratégia de vencer a fissura ler razões para ter decidido parar de fumar	Booleano	{0, 1}
Se escolheu como estratégia de vencer a fissura realizar relaxamento	Booleano	{0, 1}
Se escolheu como estratégia de vencer a fissura comer alimentos pouco calóricos	Booleano	{0, 1}
Se escolheu como estratégia de vencer a fissura beber água lentamente	Booleano	{0, 1}

4.1.2 Métricas de Similaridade

Sendo cada amostra de um conjunto de dados representada por um ponto no espaço n -dimensional é possível agrupar amostras próximas e que mais se assemelham. Durante um agrupamento a proximidade entre duas amostras pode ser indicada por um tipo de distância. Por outro lado, os itens são agrupados baseando-se em medidas de similaridade.

Existem diferentes formas de se calcular a similaridade ou dissimilaridade. Nesse trabalho foram considerados a distância euclideana, a distância de Manhattan e similaridade do cosseno.

A distância euclideana é simplesmente a distância geométrica no espaço multidimensional. A Equação (4.1) descreve a distância euclidiana entre as amostras x e y , onde x_f e y_f são os f -ésimos atributos de x e y , onde x e y são amostras pertencentes a um conjunto de dados analisado. Quanto menor a distância euclidiana entre duas amostras, mais similares elas são.

A Equação (4.2) descreve a distância de Manhattan entre as amostras x e y . A distância de Manhattan é a soma das diferenças absolutas das coordenadas de x e y . Assim como

na distância euclideana, quanto menor a distância de Manhattan entre duas amostras mais similares elas são.

A Equação (4.3) descreve a similaridade do cosseno entre as amostras x e y , definidas como vetores de atributos. A similaridade do cosseno está definida no intervalo $[-1, 1]$ e quanto mais próximo de 1 mais similares as amostras são.

$$Dist(x, y) = \sqrt{\sum_{f=1}^d (x_f - y_f)^2} \quad (4.1)$$

$$Dist(x, y) = \sum_{f=1}^d |x_f - y_f| \quad (4.2)$$

$$Sim(x, y) = \cos \theta = \frac{x \cdot y}{\|x\| \times \|y\|} \quad (4.3)$$

4.1.3 Algoritmos de Agrupamento

O *K-means*, apresentado pelo Algoritmo 1, classifica um conjunto de dados em uma quantidade de grupos pré-determinada. O objetivo é definir k centroides - um para cada grupo. O algoritmo associa cada item ao centroide mais próximo. Quando todos os itens possuem um grupo, segue iterativamente melhorando os k centroides. O algoritmo finaliza quando não há mais mudanças de itens para os grupos (MACQUEEN et al., 1967).

O algoritmo necessita de um parâmetro (k número de grupos) definido pelo usuário. Como não se sabe o número de grupos isso acaba sendo um problema, já que o número de grupos escolhido pode não representar o número de grupos que realmente existem, for-

çando alguns dados a pertencerem a grupos ao quais não se encaixam.

Algoritmo 1: K-MEANS

Entrada: Quantidade de grupos (k), dados a serem agrupados

Saída: Dados divididos na quantidade de grupos informado na entrada

1 **início**

2 | Escolhe aleatoriamente k valores para serem os centroides dos grupos;

3 | Associar cada item ao centroide mais próximo;

4 | Recalcular o centro de cada grupo;

5 | Repetir os passos até nenhum dos itens mudarem de grupo;

6 **fim**

O agrupamento hierárquico, apresentado no Algoritmo 2, particiona um número N de amostras em k grupos. O algoritmo agrupa itens de modo que duas amostras agrupadas em algum momento sempre estarão juntas, mesmo que em um próximo passo estejam em um grupo mais genéricos. Dessa forma é possível analisar os grupos em diferentes níveis de granularidade (CAMARGOS; NIETTO; NICOLETTI, 2016).

Dado um conjunto de dados D , cada item desse grupo será considerado um grupo unitário. Tais grupos são agrupados de acordo com a similaridade, até que todos os itens pertençam a um único grupo - técnica conhecida como *bottom-up*. Esse algoritmo induz agrupamentos hierárquicos aninhados. (CAMARGOS; NIETTO; NICOLETTI, 2016).

Algoritmo 2: AGRUPAMENTO HIERÁRQUICO AGLOMERATIVO

Entrada: Quantidade de grupos (k), dados a serem agrupados, métrica de distância

Saída: Dados divididos na quantidade de grupos informado na entrada

1 **início**

2 | Fazer um grupo para cada amostra;

3 | Procurar pares de grupos similares utilizando a medida de distância desejada;

4 | Associar os grupos unidos no passo anterior em grupos maiores;

5 | Recalcular a distância do grupo para todos os outros itens;

6 | Repetir os passos até existir o número de grupos desejado;

7 **fim**

4.1.4 Métrica de Qualidade

O coeficiente de silhueta, descrita através da Equação 4.4 mede o quão bem uma amostra x corresponde ao agrupamento a qual pertence, onde a é a dissimilaridade média entre x e cada amostra do grupo ao qual x pertence e b é a dissimilaridade média entre x e cada amostra do grupo mais próximo.

Os valores da silhueta podem variar entre -1 e 1, quanto mais próximo de um mais preciso é o grupo (ROUSSEEUW, 1987). Este método é utilizado no trabalho para verificar a qualidade dos grupos gerados pelos algoritmos de agrupamento escolhidos.

$$s(x) = \frac{a - b}{\max(a, b)} \quad (4.4)$$

4.2 Sistema de Recomendação

Sistemas de recomendação tem como função analisar dados de usuários e objetos e com base nessas informações ser capaz de realizar predições de um item. Existem várias formas de se implementar um sistema de recomendação.

O sistema de recomendação⁷ desenvolvido para o site Viva Sem Tabaco utiliza uma abordagem colaborativa baseada em usuários para recomendar dicas (ROCHA, 2017). Com o intuito de aprimorar a recomendação implementamos um sistema de *feedbacks* dos usuários.

O sistema de *feedbacks* está disponibilizado pelo sistema de aplicativo Viva Sem Tabaco, apresentado na Seção 3. Nele o usuário informa se gostou ou não de uma determinada dica recomendada.

O sistema de recomendação não realizava a recomendação dos dado. Para ser possível enviar as dicas recomendadas para cada usuário o sistema de recomendação sofreu algumas alterações:

- Agora os dados dos usuários e das classificações das dicas são lidos a partir do banco de dados. Anteriormente os dados eram lidos a partir de um arquivo csv, que precisava ser gerado a partir dos dados de banco de dados.

⁷Código disponível em <<https://github.com/FernandaNunesDutra/recommender-system>>

- Ao realizar uma recomendação, o sistema armazena as dicas recomendadas pelo sistema no banco de dados, juntamente com o algoritmo de agrupamento e métrica de similaridade utilizados.
- Adotou-se recomendações aleatórias para solucionar o problema de partida fria, decorrente da ausência de dados relacionados aos usuários e suas preferências em relação às dicas. Ou seja, quando não houver dados suficientes sobre as dicas e os usuários serão utilizadas recomendações aleatórias.
- A última se refere à forma como o sistema de recomendação é solicitado. Para disponibilizar o envio, o *script python* passou a ser uma API Rest, assim como o servidor. Quando o servidor receber uma requisição para obter as dicas, ele solicita ao sistema de recomendação, desenvolvido em python, a recomendação através da url `/tips/userId`. Atualmente é retornado para o servidor o identificador de dica a cada solicitação.

Além disso, os algoritmos de agrupamento e as métricas utilizadas foram sorteadas aleatoriamente a cada vez que o sistema de recomendação é chamado.

A fim de ficar disponível para futuras análises o desempenho de cada algoritmo de agrupamento e métrica utilizados, após ser realizada a recomendação são guardados no banco de dados informações da recomendação feita de acordo com os dados mostrados na Tabela 4.2.

Tabela 4.2: Dados sobre as recomendações salvas no banco

Campos
Id da dica recomendada
Id do usuário
Algoritmo de agrupamento utilizado para gerar essa recomendação
Métrica utilizada para calcular as distâncias dos pontos
Data que a recomendação foi realizada

Vale ressaltar que ao SR realizar a recomendação, ele salva no banco de dados, mas não recomenda efetivamente ao usuário. A recomendação acontece de fato quando o servidor responde a requisição feita pelo aplicativo do usuário. O aplicativo efetua uma requisi-

ção ao servidor que, por sua vez, realiza uma requisição ao sistema de recomendação e este realiza a recomendação unicamente para o usuário solicitado.

O usuário terá acesso às dicas que foram recomendadas quando acessar o aplicativo na aba dica.

4.2.1 Itens Recomendados

Os itens recomendados serão as dicas disponíveis no Apêndice A. Elas são avaliadas pelo usuário através do aplicativo com “gostei” ou “não gostei”. As avaliações “gostei” são representadas pelo valor +1, as avaliações “não gostei” são representadas pelo valor -1. As avaliações são utilizadas para calcular as recomendações (ROCHA, 2017), como mostrado na seção 4.2.2.

4.2.2 Cálculo das Predições e Recomendações

O sistema de recomendação implementado utiliza uma abordagem colaborativa baseada em usuários (ROCHA, 2017). A FC é popular e realiza suas predições baseada nas classificações, comportamentos e perfis dos usuários (EKSTRAND; RIEDL; KONSTAN, 2010).

Durante as classificações pode-se utilizar o valor da similaridade para gerar previsões usando média ponderada. Os itens são recomendados escolhendo o itens com previsões mais altas. O cálculo da predição de um item i para um usuário u é descrito pela Equação 4.5 (EKSTRAND; RIEDL; KONSTAN, 2010).

$$P(u, i) = \frac{\sum_{v \in V} sim(u, v) \cdot r_{v,i}}{\sum_{v \in V} |sim(u, v)|} \quad (4.5)$$

$r(v, i)$ é a avaliação do item i pelo usuário v . $sim(u, v)$ é a similaridade entre os usuários u e v . V é o conjunto de vizinhos (usuários que pertence ao mesmo grupo) do usuário u (EKSTRAND; RIEDL; KONSTAN, 2010).

Uma predição com valor positivo significa um item que o usuário irá gostar. Já uma predição com valor negativo, indica o contrário, um item que provavelmente o usuário não irá gostar (EKSTRAND; RIEDL; KONSTAN, 2010).

4.2.3 Execução do Sistema de Recomendação

O Sistema de recomendação proposto realiza um número de passos para indicar uma recomendação ao usuário:

1. Sortear aleatoriamente um algoritmo de agrupamento;
2. Sortear aleatoriamente uma métrica caso o algoritmo de agrupamento sorteado seja o hierárquico aglomerativo, caso o sorteado seja o K-means a distância utilizada será a euclideana
3. Ler dados dos usuários e das dicas e exportar para dois arquivos locais;
4. Agrupar usuários no número de grupos enviados por parâmetro;
5. Realizar as predições, fórmula apresentada na Seção 4.2.2, para todas as dicas avaliadas por um vizinho do usuário;
6. Recomendar a dica com maior predição.

5 Experimentos

Foram realizados alguns experimentos no projeto com o intuito de verificar a utilização do aplicativo e o funcionamento das recomendações. Além disso, tem como objetivo analisar o melhor número de grupos para a base de dados de testes..

Diferentes sistemas de recomendações de modelo colaborativo fazendo combinações entre algoritmos de agrupamento e métricas foram propostos (ROCHA, 2017):

- filtragem colaborativa + K-means + distância euclidiana.
- filtragem colaborativa + hierárquico aglomerativo + distância euclidiana.
- filtragem colaborativa + hierárquico aglomerativo + distância de Manhattan.
- filtragem colaborativa + hierárquico aglomerativo + similaridade do cosseno.

A estratégia aleatória foi escolhida para ser possível coletar informações sobre o agrupamentos e métricas. Dessa forma, será possível analisar os dados sobre o desempenho de cada agrupamento. Caso o sorteado seja o *K-means* utiliza-se ele combinado com a distância euclidiana, caso o algoritmo de agrupamento sorteado seja o hierárquico aglomerativo, o SR sorteia entre distância euclidiana, distância de Manhattan e similaridade do cosseno.

Foi decidido que será recomendada apenas uma dica por vez para se ter mais informações sobre as avaliações na próxima recomendação. Entretanto é possível escolher o número de dicas a serem recomendadas apenas alterando uma variável global do sistema de recomendação desenvolvido.

5.1 Base de Dados

A base de dados utilizada para realizar os experimentos foi a base de dados do VST. A base de dados possui um cadastrado de todas as 47 dicas existentes, apresentadas no Apêndice

A, os usuários do sistema e as informações das avaliações feitas pelos usuários do aplicativo. Pretende-se aumentar a base de dados, tanto em dicas registradas, quanto de usuários utilizando a intervenção.

Por não haver dados sobre as avaliações das dicas dos usuários do VST, foi utilizada uma base de teste. Foram cadastrados dez usuários que utilizaram o aplicativo por uma semana para ser possível coletar informações sobre eles e utilizá-las para realizar recomendações.

Os teste foram realizados com a base dados apenas com as informações dos usuários cadastradas (*e-mail*, senha e data de nascimento), além dos dados descritos na Tabela 4.1 e das dicas.

5.1.1 Características dos Usuários

Os usuários que realizaram os testes possuem as características apresentadas nos gráfico das Figuras 5.1, 5.2 e 5.3.

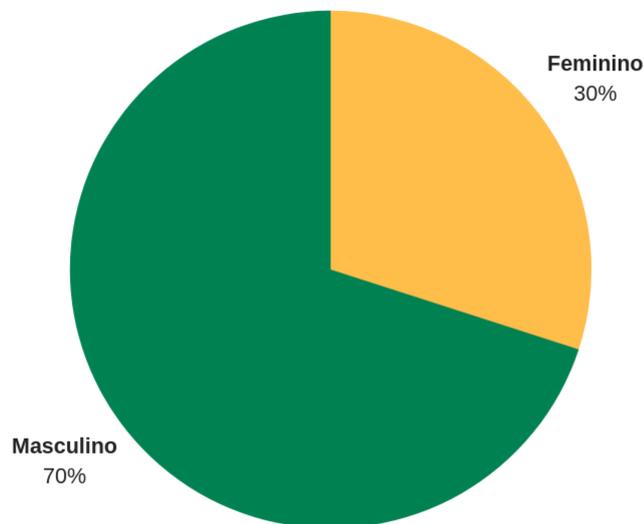


Figura 5.1: Gráfico de Pizza do sexo dos usuários testadores.

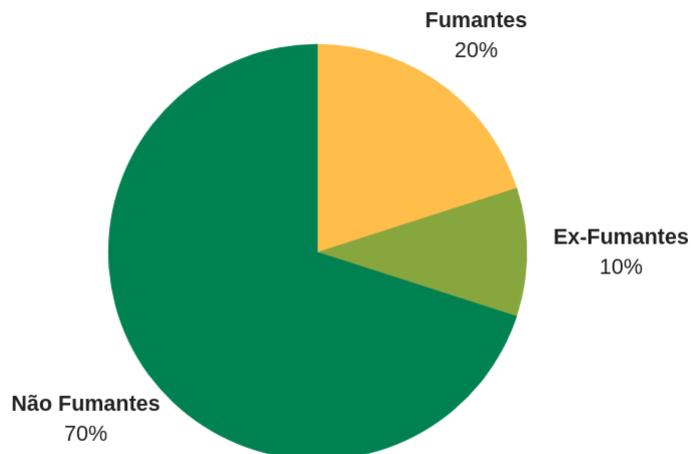


Figura 5.2: Gráfico de Pizza com o número de usuários fumantes, ex-fumantes e não fumantes.

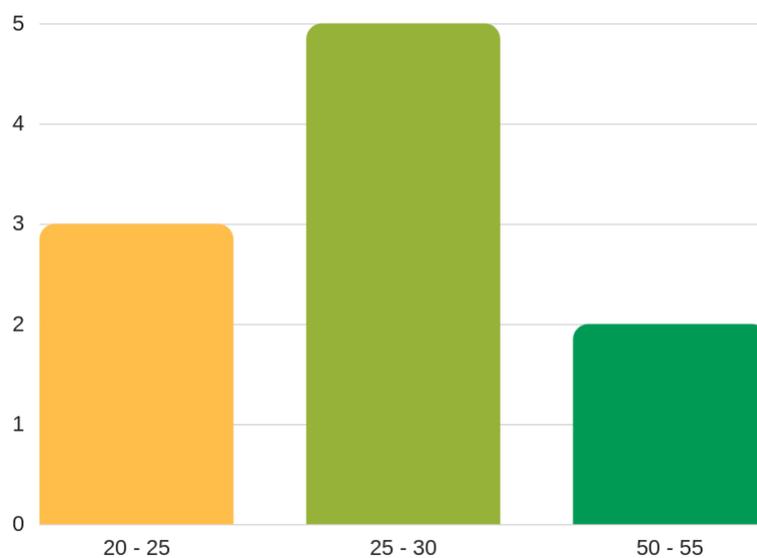


Figura 5.3: Gráfico de barra da idade dos usuário testadores.

Como apresentado no gráfico da Figura 5.1, a maioria dos usuários que realizaram são do sexo masculino. Apesar do aplicativo ser voltado para usuários fumantes grande parte responsável por testar o SR foram Não fumantes e apenas alguns com contato com o tabagismo (2 fumantes e 1 ex-fumante), como mostra no gráfico da Figura 5.2. Através do gráfico da Figura 5.3 pode-se observar que metade dos testadores possuem entre 25 a 30 anos, a outra metade está distribuída entre 20 e 25 anos, com 3 usuários, entre 50 e 55 anos, com 2 usuários.

O objetivo dos experimentos é avaliar o funcionamento do aplicativo juntamente com o sistema de recomendação e o servidor desenvolvido, por esse motivo a base de teste não foi focada exclusivamente em usuários fumantes.

5.2 Experimento 1

O experimento 1 foi realizado com o objetivo de analisar a qualidade dos grupos gerados pelos algoritmos, para isso foi utilizado o coeficiente de silhueta, apresentado na Seção 4.1.4. Foram realizadas dez execuções independentes de todos os algoritmos combinados com as métricas. A Tabela 5.1 apresenta a média do coeficiente de silhueta encontrado.

Tabela 5.1: Resultado do coeficiente silhueta para cada algoritmo e métrica

Nº Grupos	K-means	Aglomerativo + distância euclidiana	Aglomerativo + distância Manhattan	Aglomerativo + similaridade do cosseno
2	0.3324	0.3330	0.5251	0.4416
3	0.2700	0.2702	0.4239	0.5450
4	0.1331	0.2227	0.3527	0.4017
5	0.1127	0.1184	0.2164	0.1572

Como mostrado na Tabela 5.1, em geral os algoritmos têm o valor de coeficiente de silhueta maior com apenas dois grupos. Com exceção do hierárquico aglomerativo combinado com a similaridade do cosseno que possui melhor desempenho com três grupos. Apesar disso, ele apresenta um resultado bom para dois grupos.

Devido ao resultado apresentado na Tabela 5.1, os próximos experimentos serão re-

alizados com apenas dois grupos, já que todos os algoritmos e métricas em geral possuem valor do coeficiente maior com dois grupos. Apesar disso, todos os agrupamentos obtiveram o coeficiente de silhueta positivo, o que mostra que estão bem definidos segundo esse critério. E com o resultados apresentados na Tabela 5.1, podemos dizer que o melhor desempenho é o hierárquico aglomerativo utilizando a similaridade do cosseno como métrica, utilizando 3 grupos.

5.3 Experimento 2

O experimento 2 tem como objetivo analisar as recomendações aleatórias realizadas na primeira fase. Nos primeiros dois dias foram recomendados apenas dados aleatórios, para coletar dados sobre as avaliações dos usuários. Realizou-se 20 recomendações aleatórias durante o período de teste. Cada usuário recebeu aproximadamente 2 dicas distintas como recomendação.

Não foram cobertas todas as dicas cadastradas, por não terem sido recomendadas aleatoriamente, essas dicas acabaram nunca sendo recomendadas pelo SR, já que ele utiliza as avaliações dos usuários para recomendar.

As avaliações obtidas nos experimentos aleatórios são apresentadas no gráfico da Figura 5.4. A dica “Em quase 3 semanas tente fazer um planejamento futuro e procure imaginar sua vida sem o cigarro e quais os benefícios isso trará para você” foi indicada aleatoriamente para 3 usuários diferentes e todos eles classificaram as dicas como “não gostei”. Ao todo foram 6 dicas avaliadas negativamente e 14 avaliadas positivamente.

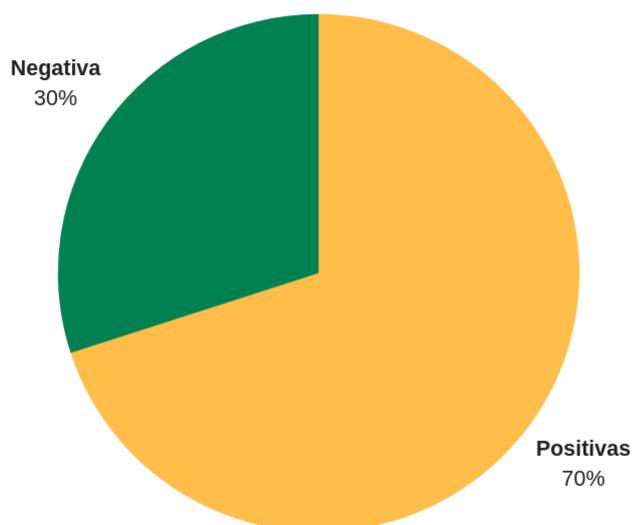


Figura 5.4: Gráfico de Pizza das avaliações dos usuários para as dicas indicadas aleatoriamente.

As dicas aleatórias tiveram um desempenho positivo, como mostra no gráfico 5.4, apesar de terem um número significativo de avaliações negativas.

5.4 Experimento 3

O experimento 3 tem como objetivo analisar o sistema de recomendação funcionando após já haver algumas recomendações aleatórias avaliadas disponíveis no banco de dados.

No segundo dia de testes já era possível utilizar o SR para recomendar, baseando-se nas avaliações feitas pelos usuários sobre as dicas recomendadas aleatoriamente no dia anterior.

É importante ressaltar que apesar de ser possível recomendar através do sistema de recomendação não deve-se abandonar as recomendações aleatórias por completo, já que é necessário ter avaliações das dicas para que elas possam ser recomendadas. Utilizar o SR agora limitaria a recomendação em apenas as dicas recomendadas aleatoriamente e avaliadas pelos usuários. Para contornar esse problema, enquanto houver dicas não avaliadas

serão indicadas as dicas ainda não avaliadas aleatoriamente .

Foram feitas em média duas recomendações através do SR para cada usuário. Pode-se ver os resultados das avaliações das dicas recomendadas pelo SR no gráfico da Figura 5.5.

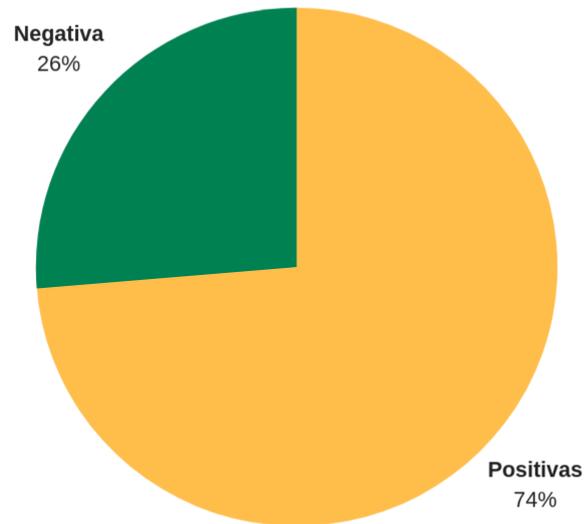


Figura 5.5: Gráfico de Pizza das avaliações dos usuários para as dicas indicadas pelo SR.

Devido ao número restrito de recomendações observou-se que os usuários de um dos grupos receberam em geral as mesmas dicas “Foque sua atenção em coisas que não estejam relacionadas ao desejo de fumar” e “A fissura dura em média de 2 a 5 minutos. Nesses momentos beba um copo de água ou escove os dentes”. Durante análise do SR, verificou-se que essas eram as dicas com maiores predições para grande parte dos usuários, e teve avaliações positivas em todas suas indicações, por isso ela acabou sendo indicada diversas vezes para diferentes usuários.

Os experimentos 2 e 3 mostraram que não houve grandes diferenças entre a recomendação aleatória e a recomendação realizada pelo SR. Apesar dos testes mostrarem uma pequena diferença, é possível que essa diferença seja maior, já que os dados testados eram limitados e não havia uma grande massa de dados para o SR trabalhar e poder realizar uma recomendação melhor.

Acredita-se que com o uso do aplicativo *mobile* e o consequente aumento no nú-

mero de dados no banco de dados, o SR irá melhorar seu desempenho em comparação as recomendações aleatórias.

6 Conclusões e Trabalhos Futuros

No projeto foram desenvolvidos dois sistemas o aplicativo *mobile* do VST e o servidor. O aplicativo foi um aprimoramento do aplicativo desenvolvido anteriormente, ele traz um *layout* mais amigável e maior usabilidade, com dicas disponíveis para serem avaliadas pelos usuários. O servidor foi desenvolvido como um sistema auxiliador do aplicativo, para que o aplicativo pudesse interagir com os dados do banco de dados. Além disso, foram feitas alterações no SR para que ele ser capaz de se adaptar ao novo ambiente e recomendar dicas.

O SR mostrou-se com potencial para indicar boas dicas quando houver mais informações sobre as avaliações das dicas. O aplicativo melhorou muito se comparado ao aplicativo desenvolvido anteriormente, já que a versão desenvolvida para esse trabalho possui a persistência dos dados e permite o cadastro simples de novas dicas e desafios. O sistema de recomendação desenvolvido para a intervenção do Viva Sem Tabaco quando combinado com a intervenção através de um aplicativo de celular possui grandes chances de melhorar o processo de cessação de tabaco.

Com o uso do aplicativo e o sistema de *feedback*, o SR poderá aprender o que é negativo para o usuário e assim poder realizar melhores recomendações. Acredita-se que haverá um melhor desempenho do SR quando existir um número satisfatório de avaliações das dicas.

Além disso, com uma intervenção mais acessível, através do celular, e um sistema de recomendação indicando dicas para o seu perfil é provável que mais usuários prossigam no processo de cessação do uso de tabaco.

Como trabalhos futuros as proposta são o estudo e avaliação do sistema de recomendação de dicas e seu comportamento após um número significativo de *feedbacks* dos usuários, desenvolver um sistema de recomendação utilizando outro algoritmo de agrupamento como DBScan e compará-lo com os algoritmos utilizados atualmente. Outra proposta para os trabalhos futuros é aplicar a nova versão do aplicativo os elementos de *gameficação* e permitir o cadastro de usuários através o aplicativo e via *facebook*, disponibilizar o aplicativo publicamente para ser utilizado pelos usuários do VST, permitir que os usuá-

rios possam inserir dicas que estão funcionando para ele durante o tratamento. Além disso, avaliar a eficácia do aplicativo quanto à manutenção da interação do usuário com o sistema quando comparado com a intervenção via web, avaliar a eficácia do aplicativo quanto à permanência do usuário sem fumar, incorporar novas funções no aplicativo, tais como um chat com profissionais de psicologia que possam auxiliar os usuários.

Bibliografia

- ADOMAVICIUS, G.; TUZHILIN, A. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE transactions on knowledge and data engineering*, IEEE, v. 17, n. 6, p. 734–749, 2005.
- BALABANOVIĆ, M.; SHOHAM, Y. Fab: content-based, collaborative recommendation. *Communications of the ACM*, ACM, v. 40, n. 3, p. 66–72, 1997.
- BENNETT, J.; LANNING, S. et al. The netflix prize. In: NEW YORK, NY, USA. *Proceedings of KDD cup and workshop*. [S.l.], 2007. v. 2007, p. 35.
- BOBADILLA, J. et al. Recommender systems survey. *Knowledge-based systems*, Elsevier, v. 46, p. 109–132, 2013.
- BRAGA, A. d. P.; CARVALHO, A.; LUDERMIR, T. B. *Redes neurais artificiais: teoria e aplicações*. [S.l.]: Livros Técnicos e Científicos, 2000.
- BURKE, R. Hybrid recommender systems: Survey and experiments. *User modeling and user-adapted interaction*, Springer, v. 12, n. 4, p. 331–370, 2002.
- CAMARGOS, R. C.; NIETTO, P. R.; NICOLETTI, M. do C. Agglomerative and divisive approaches to unsupervised learning in gestalt clusters. In: *ISDA*. [S.l.: s.n.], 2016.
- EKSTRAND, M. D.; RIEDL, J. T.; KONSTAN, J. A. Collaborative filtering recommender systems. In: *Foundations and Trends in Human–Computer Interaction*. [S.l.]: Now Publishers Inc, 2010. v. 4, p. 81—173. ISBN 978-1-60198-442-5.
- FENSEL, D.; BUSSLER, C. The web service modeling framework wsmf. *Electronic Commerce Research and Applications*, Elsevier, v. 1, n. 2, p. 113–137, 2002.
- FUNG, G. A comprehensive overview of basic clustering algorithms. 2001.
- GOMIDE, H. P. et al. Development of an open-source web-based intervention for Brazilian smokers – Viva sem Tabaco. *BMC Medical Informatics and Decision Making*, v. 16, p. 103, 2016. ISSN 1472-6947. Disponível em: <<http://dx.doi.org/10.1186/s12911-016-0339-7>>.
- Instituto Nacional do Câncer. *Abordagem e tratamento do fumante - consenso 2001*. Rio de Janeiro: INCA, 2001.
- LANGLEY, P.; SIMON, H. A. Applications of machine learning and rule induction. *Communications of the ACM*, ACM, v. 38, n. 11, p. 54–64, 1995.
- LINDEN, R. Técnicas de agrupamento. *Revista de Sistemas de Informação da FSMA*, v. 1, n. 4, p. 18–36, 2009.
- LU, J. et al. Recommender system application developments: a survey. *Decision Support Systems*, Elsevier, v. 74, p. 12–32, 2015.
- MACQUEEN, J. et al. Some methods for classification and analysis of multivariate observations. In: OAKLAND, CA, USA. *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*. [S.l.], 1967. v. 1, n. 14, p. 281–297.

- METEREN, R. V.; SOMEREN, M. V. Using content-based filtering for recommendation. In: *Proceedings of the Machine Learning in the New Information Age: MLnet/ECML2000 Workshop*. [S.l.: s.n.], 2000. p. 47–56.
- MITCHELL, T. M. Machine learning and data mining. *Communications of the ACM*, ACM, v. 42, n. 11, p. 30–36, 1999.
- NETO, J. M.; MOITA, G. C. Uma introdução à análise exploratória de dados multivariados. *Química nova*, SciELO Brasil, v. 21, n. 4, p. 467–469, 1998.
- NG, R. T.; HAN, J. Efficient and effective clustering methods for spatial data mining. In: *Proc. of*. [S.l.: s.n.], 1994. p. 144–155.
- PAI, M. V. Restful web services. Citeseer, 2014.
- PARDO, T. A. S.; NUNES, M. d. G. V. Aprendizado bayesiano aplicado ao processamento de línguas naturais. *Série de Relatórios Técnicos do Instituto de Ciências Matemáticas e de Computação-ICMC, Universidade de São Paulo*, n. 180, 2002.
- PELLUCCI, P. R. S. et al. Utilização de técnicas de aprendizado de máquina no reconhecimento de entidades nomeadas no português. *e-xacta*, v. 4, n. 1, 2011.
- PEREIRA, D. A. *Uso de Gamificação numa Intervenção via Aplicativo Móvel para a Cessação do Consumo de Tabaco*. 2017.
- RESNICK, P. et al. Grouplens: an open architecture for collaborative filtering of netnews. In: ACM. *Proceedings of the 1994 ACM conference on Computer supported cooperative work*. [S.l.], 1994. p. 175–186.
- ROBERTS, A.; BROOKS, R.; SHIPWAY, P. Internal combustion engine cold-start efficiency: A review of the problem, causes and potential solutions. *Energy Conversion and Management*, Elsevier, v. 82, p. 327–350, 2014.
- ROCHA, T. C. R. D. *Sistemas de Recomendação Para Intervenções do Consumo de Álcool e Tabaco Utilizando Técnicas de Agrupamento*. 2017. Available at <http://monografias.ice.ufjf.br/tcc-web/tcc?id=288>.
- RODRIGUEZ, A. Restful web services: The basics. *IBM developerWorks*, v. 33, 2008.
- ROUSSEEUW, P. J. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, Elsevier, v. 20, p. 53–65, 1987.
- SATHYA, R.; ABRAHAM, A. Comparison of supervised and unsupervised learning algorithms for pattern classification. *International Journal of Advanced Research in Artificial Intelligence*, v. 2, n. 2, p. 34–38, 2013.
- SCHEIN, A. I. et al. Methods and metrics for cold-start recommendations. In: ACM. *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*. [S.l.], 2002. p. 253–260.
- SOUTO, M. de et al. Técnicas de aprendizado de máquina para problemas de biologia molecular. *Sociedade Brasileira de Computação*, 2003.
- U.S Department of Health Services. *Treating Tobacco Use and Dependence: 2008 Update*. [S.l.: s.n.], 2008.

XU, R.; WUNSCH, D. Survey of clustering algorithms. *IEEE Transactions on neural networks*, IEEE, v. 16, n. 3, p. 645–678, 2005.

YAO, G.; CAI, L. User-based and item-based collaborative filtering recommendation algorithms design. *University of California, San Diego*, 2017.

ZHU, R.; GONG, S. Analyzing of collaborative filtering using clustering technology. In: IEEE. *Computing, Communication, Control, and Management, 2009. CCCM 2009. ISECS International Colloquium on*. [S.l.], 2009. v. 4, p. 57–59.

A Apêndice Dicas Recomendadas

- A fissura dura em média de 2 a 5 minutos. Nesses momentos beba um copo de água ou escove os dentes.
- Escovar os dentes após comer ou beber algo que você associa ao cigarro pode diminuir sua vontade de fumar. Quando sentir vontade de acender um cigarro coma alimentos de baixo de teor calórico como por exemplo, frutas, cenouras, salsão, pepino, etc.
- Durante o processo de cessação é comum ocorrer episódios de recaída, mas não desanime, continue firme no seu propósito de parar de fumar.
- Comer frutas ou alimentos de baixas calorias ajudam nos momentos de fissura e a não ganhar peso.
- Parar de fumar aumenta a expectativa e a qualidade de vida.
- Tenha sempre perto de você garrafas de água e quando tiver vontade de fumar tome alguns goles da água.
- Ao parar de fumar você se sentirá melhor fisicamente e terá mais disposição para realizar exercícios físicos.
- Ao deixar de fumar você reduz as chances de desenvolver doenças do coração, doenças pulmonares e cânceres.
- Se além de você existe outro fumante em casa, peça para que ele evite fumar perto de você e proponha a essa pessoa uma tentativa de parar. Um poderá dar suporte ao outro.
- Guarde o dinheiro que você gasta com cigarros e ao final de um ano você poderá investir algo prazeroso, como uma viagem, por exemplo.
- Evite comidas ou bebidas que você associa ao cigarro, como por exemplo, café, álcool, doces, etc.

- Pense nas suas atividades diárias e em que momento você tem vontade de fumar e procure mudar esse comportamento.
- Participar de grupos de apoio para fumantes/ex-fumantes podem aumentar suas chances de parar de fumar.
- Em 15 anos após parar, os riscos de sofrer de doenças cardíacas são iguais a de uma pessoa que nunca fumou.
- Não desista diante das recaídas. A maioria das pessoas que conseguem deixar de fumar não conseguem da primeira vez.
- Comunique seus familiares e amigos próximos sobre seu processo de parar de fumar e peça a compreensão deles durante esse período.
- Identifique as situações que te levam a fumar e procure evitá-las.
- Ao invés de acender um cigarro quando você estiver estressado, respire profundamente durante alguns minutos e tente relaxar.
- Procure o apoio de sua família e amigos se você estiver precisando desabafar.
- Evite frequentar ambientes onde todos fumam para não despertar o desejo de acender um cigarro.
- A prática de exercícios físicos pode ajudar a diminuir a ansiedade e evitar o ganho de peso.
- Procure evita situações estressantes durante o processo de parar de fumar.
- Beba bastante água durante o dia para ajudar na desintoxicação da nicotina.
- Tente mudar sua rotina para evitar situações associadas ao hábito de fumar.
- Faça uma lista de razões pelas quais é importante você parar de fumar.
- Procure o apoio de grupos de pessoas que também estão tentando parar de fumar. Há grupos de ajuda na internet, inclusive *Facebook*

- Evite comportamentos ou situações que reforcem o desejo de acender um cigarro.
- Em 1 dia sem fumar, você já deixou de inalar mais de 4.700 substâncias tóxicas, tais como a amônia, monóxido de carbono, acetona e alcatrão.
- Em 2 dias sem fumar, sua pressão sanguínea e pulsação estão iguais a de um não fumante.
- Em 3 dias sem fumar seu olfato e paladar já começam a ganhar sensibilidade novamente.
- No 4^a dia sem fumar o nível de oxigênio no seu sangue se normaliza e você não tem mais nicotina circulando no seu sangue.
- Em 5 dias sem fumar seus brônquios estão funcionando melhor e estão limpando os resíduos deixados pelo fumo nas vias respiratórias.
- Nos primeiros dias sem cigarros a vontade fumar é maior, procure frequentar lugares onde uso do cigarro é proibido como cinemas, teatros, shoppings por exemplo.
- Em 7 dias sem fumar sua pele e cabelo já devem estar melhores.
- Em 8 dias sem fumar e você pode se sentir um pouco irritado. Por isso tente fazer exercícios de relaxamento ou fazer algo que lhe traga bem-estar.
- Em 9 dias sem fumar sua boca e pele não tem mais aquele cheiro forte produzido pelo cigarro.
- Nas primeiras semanas sem fumar você pode se sentir muito ansioso ou nervoso. Nesses momentos fique de pé, feche os olhos e respire lenta e profundamente. Isso ajudará a se relaxar.
- Em 11 dias sem acender um cigarro. Procure realizar alguma atividade física. Ninguém sente vontade de fumar enquanto corre, anda de bicicleta ou nada.
- No 12^a dia sem fumar o forte desejo de fumar vai se tornando mais brando.
- Em quase 2 semanas sem cigarros você se cansará menos ao fazer exercícios físicos.

-
- Mantenha-se confiante no seu propósito de parar de fumar. Acredite em você!
 - Em 15 dias sem fumar sua respiração fica mais fácil e você se sente mais disposto para realizar suas atividades.
 - Após 2 semanas seu apetite vai melhorar, seus sentidos de gostos e olfato também. Os alimentos terão melhor sabor e aroma.
 - Em quase 3 semanas tente fazer um planejamento futuro e procure imaginar sua vida sem o cigarro e quais os benefícios isso trará para
 - você.
 - Foque sua atenção em coisas que não estejam relacionadas ao desejo de fumar.
 - Faça o cálculo do quanto de dinheiro você já economizou sem fumar.
 - Resista ao desejo de acender um cigarro! A vontade passa tão rápida quanto chega.