Escola Politécnica da Universidade de São Paulo

PMR2550 – Projeto de Conclusão do Curso II

Desenvolvimento de um Sistema de Recomendação para e-Commerces

Nome	*Número USP*
Antônio Guilherme Ferreira Viggiano	6846450
Fernando Fochi Silveira Araújo	5894546

Orientador

Prof. Dr. Fábio Gagliardi Cozman

16 de setembro de 2014

Lista de símbolos

k	Número de vizinhos mais próximos
N	Tamanho da lista de recomendação
$\mathcal{U}$	Conjunto de todos os usuários
$\mathcal{I}$	Conjunto de todos os itens
$\mathcal{F}$	Conjunto de todos os atributos dos itens
$\mathcal{C}$	Conjunto de todas as características dos usuários
u, v	Usuários
i, j	Itens
f	Atributos dos itens
c	Características dos usuários
$\mathbf{X}_{M\times N}$, $\mathbf{X}$	Matriz de elementos x_{mn}
$\mathbf{x}_N$, $\mathbf{x}$	Vetor de elementos x_n
$\tilde{x}$	Valor ótimo de x
$\hat{x}$	Valor estimado de x
$\lvert\mathcal{X}\rvert$	Número de elementos do conjunto $\mathcal{X}$
$\mathbf{R}, r_{ui}$	Avaliação feita pelo usuário u do item i
$\mathbf{A}, a_{if}$	Atributo f presente no item i
$\mathbf{B}, b_{uc}$	Característica c do usuário u
$\mathbf{T}, t_{uf}$	Correlação entre usuário u e atributo f
$\mathbf{S}, s_{ij}, s_{uv}$	Similaridade entre itens i e j ou entre usuários u e v
$\mathbf{W}, w_{uf}$	Correlação ponderada entre usuário u e atributo f
$\mathbf{w}, w_f$	Peso do atributo f

Sumário

1 Introdução

"Sistemas de recomendação são ferramentas e técnicas de software destinadas a prover sugestões de itens para usuários" (1). O sistema tem o propósito de automatizar o processo de recomendação e auxiliar na tomada de decisão, podendo ser aplicado em diversas áreas da indústria, tais como na indicação de notícias, músicas, relações de amizade ou artigos científicos.

De modo geral, um sistema de recomendação possui três etapas: a aquisição dos dados de entrada, a determinação das recomendações e finalmente a apresentação dos resultados ao usuário. A aquisição dos dados de entrada pode ser feita tanto de forma automática quanto manual, e em geral utiliza-se um banco de dados para armazenar essas informações. As sugestões são feitas segundo uma estratégia de recomendação determinada a priori, que pode ser fundamentada nas preferências do usuário, nas características dos itens ou em alguma formulação mista. Finalmente, os resultados são apresentados na interface sob variadas formas, como por exemplo em uma lista dos N itens mais relevantes para o usuário.

Conforme o tipo específico de itens recomendados, o design do sistema, a interface homem-máquina e o tipo de técnica de recomendação são construídos a fim de prover sugestões mais adequadas.

Os sistemas de recomendação são destinados primeiramente aos indivíduos que não possuem competência ou experiência suficiente para avaliar o grande número de opções do conjunto total de itens. Dessa forma, a interface homem-máquina é adaptada a cada um dos usuários, de maneira que eles recebam recomendações adequadas ao seu perfil. Essa ideia, amplamente divulgada por um antigo diretor executivo do e-commerce *Amazon.com*, se resume à sua fala de que "se você possui 2 milhões de clientes na web, você precisa ter 2 milhões de lojas na web" (2).

Motivados pela importância econômica crescente de lojas de varejo online, bem como pela possibilidade de criar um conjunto de ferramentas *open source* que possam ser utilizadas abertamente pela comunidade, propomos como Trabalho de Conclusão de Curso o desenvolvimento de um sistema de recomendação de produtos de e-commerces.

A contribuição científica e tecnológica do trabalho para a Engenharia Mecatrônica estão sobretudo nos campos de sistemas de informação, de automação de processos e de inteligência artificial. As competências acadêmicas necessárias para a sua execução envolvem algoritmos e estruturas de dados, aprendizado de máquina e modelagem de bancos de dados. As competências técnicas abrangem programação estatística e orientada a objetos (R ou Java, por exemplo) e em linguagem de consulta estruturada (SQL).

2 Objetivos

O objetivo do presente Trabalho de Conclusão de Curso é o desenvolvimento de um Sistema de Recomendação de produtos para e-commerces, e respectiva análise de desempenho das recomendações propostas.

Serão propostos três diferentes algoritmos de recomendação, e será feita uma avaliação comparativa entre eles. A explicação detalhada dos métodos se encontra no Capítulo 5.

Essa ferramenta tem como finalidade a automatização do processo de sugestão de itens, e pode ser aplicada em diversas áreas da indústria, tais como na indicação de notícias, músicas, relações de amizade ou artigos científicos. No nosso trabalho, o sistema terá como foco a sugestão de produtos de lojas de comércio online que disponham de um histórico de compras dos usuários e das características dos produtos.

A qualidade das recomendações será avaliada quanto a precisão e abrangência. Para os métodos em que se possui uma medida de similaridade entre produtos, será avaliada a distância entre os itens efetivamente comprados pelo cliente e aqueles previstos pelo sistema. Uma descrição detalhada da avaliação do sistema de recomendação está descrita no Capítulo 6.

Por meio de uma validação cruzada, analisaremos a influência dos principais parâmetros do problema na qualidade das recomendações, como o tamanho do banco de dados ou a quantidade de informações de itens e clientes utilizadas na recomendação.

Será discutido o impacto dos principais desafios tecnológicos e científicos dos sistemas de recomendação na nossa proposta de solução, tais como a escalabilidade, a adaptação a novos usuários e a esparsidade dos dados (3).

Ao final, será possível extrair uma validação experimental das diretrizes fundamentais a serem seguidas por e-commerces que desejem desenvolver um sistema de recomendação próprio ou que queiram utilizar o sistema desenvolvido neste trabalho.

3 Estado da Arte

As terminologias *cliente* e *usuário* neste texto serão intercambiáveis e sem distinção semântica, mesmo que na prática essas duas entidades possam ser diferentes. Da mesma forma, *item* e *produto* terão o mesmo significado neste trabalho.

A fim de tornar a formulação mais genérica, também não faremos distinção entre *avaliação positiva* de um item e *compra* de um item. Avaliação positiva é toda avaliação r_{ui} do item i feital pelo usuário u tal que $r_{ui} > M$, e avaliação negativa tal que $r_{ui} \leq M$, sendo M um valor mínimo escolhido a priori, indicador de que o usuário u "gostou" do item i. No caso de um banco de dados sem avaliações dos produtos, será levada em conta a compra dos itens e será admitida avaliação unitária e valor mínimo nulo. Desta forma, os bancos de dados que contenham informações do tipo "usuário u avaliou o item i em $r_{ui} = 3.54 > M$" e aqueles que contenham "usuário u comprou o item i, logo $r_{ui} = 1 > 0$" serão tratados equivalentemente.

3.1 Estado da arte dos problemas

O problema de recomendação pode ser formulado como se segue, adaptado da referência (4), com notação inspirada em (5):

"Seja $\mathcal{U}$ o conjunto de todos os usuários e seja $\mathcal{I}$ o conjunto de todos os itens que podem ser recomendados, tais como livros, filmes ou artigos científicos. Seja ℓ uma função de utilidade, que mede a relevância do produto i para usuário u. Em notação matemática, $\ell : \mathcal{U} \times \mathcal{I} \to \mathcal{R}$, onde $\mathcal{R}$ é um conjunto totalmente ordenado – por exemplo, números inteiros ou números reais dentro de um determinado intervalo, em geral $\{-1, 0, +1\}$ ou $[1, 5]$. O objetivo do sistema de recomendação é determinar o item $\tilde{\imath}_u$ que maximize a utilidade ℓ_{ui} do usuário u."

$$\forall u \in \mathcal{U},\ \tilde{\imath}_u = \arg\max_{i \in \mathcal{I}} \ell_{ui} \tag{3.1}$$

O problema central da recomendação é que "em geral a função ℓ é desconhecida ou não é definida para todo o espaço $\mathcal{U} \times \mathcal{I}$", e portanto determinar $\tilde{\imath}$ através da equação 3.1 é inviável.

Em algumas formulações, "a utilidade é descrita pela avaliação r_{ui} do item i feita pelo usuário u". Neste caso, o sistema de recomendação busca determinar $\hat{r}_{ui}$ que melhor se aproxime de r_{ui}, e a qualidade da recomendação é normalmente descrita pela distância

entre esses dois valores. Em outros sistemas, todavia, a utilidade é descrita diferentemente, de forma que o item com maior valor de $\hat{r}_{ui}$ não é necessariamente recomendado.

Para lidar com o problema da recomendação, existem três grandes grupos de estratégias de sugestão de itens, segundo as referências (4, 6):

- Recomendações baseadas em conteúdo: o usuário recebe sugestões de itens similares àqueles pelos quais ele se interessou no passado;
- Recomendações colaborativas: o usuário recebe sugestões de itens que pessoas com preferências semelhantes gostaram no passado;
- Recomendações híbridas: esses métodos combinam caracteríticas de sistemas colaborativos e baseados em conteúdo. O usuário recebe sugestões de itens compatíveis com seu perfil e de itens do interesse de usuários com perfil similar.

As estratégias de recomendação baseadas em conteúdo exploram os dados dos itens para calcular a sua relevância conforme o perfil do usuário. Suas técnicas de recomendação podem ser classificadas em dois grupos: aquelas baseadas em heurísticas ou memória – fazem a previsão com base em toda a coleção de itens anteriormente classificados pelos usuários – e aquelas baseadas em modelos – utilizam o conjunto de avaliações com o objetivo de descrever a interação entre usuários e itens, tal como em uma regressão linear ou em uma rede Bayesiana.

Na abordagem de sistemas baseados em conteúdo, a recomendação pode ser vista como um problema de aprendizado de máquina, em que o sistema adquire conhecimento sobre o usuário. Muitas vezes é recomendado que o aprendizado seja feito com base no perfil do usuário em uso continuo, ao invés de forçá-lo a responder diversas perguntas demográficas (7) – idade, gênero, classe social, etc. O objetivo é categorizar novas informações baseadas em informações previamente adquiridas e rotuladas como interessantes ou não pelo usuário. Com estas informações em mão, é possível gerar modelos preditivos que evoluem conforme aparecem novas informações.

Em sistemas baseados em conteúdo, os itens a serem recomendados podem possuir diversos atributos e formas de classificação. Em documentos como e-mails, *websites* ou comentários de usuários, os textos não tem estrutura definida e a abordagem mais comum para escolher o melhor item é a mineração de informações. O usuário procura por uma lista de termos desejados e o sistema retorna os textos de maior relevância, tal como é feito em um motor de busca (8). Nesses casos, calcula-se a similaridade entre documentos a partir da importância das palavras ou termos similares, como a TF-IDF ou o classificador Bayesiano (9).

Em bancos de dados relacionais, os itens possuem uma categorização pré-definida, e sua relevância depende das suas características, descritas pela matriz de atributos $\mathbf{A}$. Cada

feature pertence a um conjunto distinto, podendo ser booleano (possui ou não possui), inteiro ou real (preço, data, etc.), ou um coleção finita de valores (marca, modelo, gênero, etc.), como exemplifica a Tabela 1.

Tabela 1 – Atributos a_{if}

	f_1	f_2	f_3	f_4
i_1	1	50	0.8	P
i_2	0	75	0.3	M
i_3	1	30	0.4	G

As recomendações colaborativas, por sua vez, tentam prever a utilidade dos itens para cada cliente com base em itens previamente avaliados por outros usuários. Elas podem ser baseadas em usuários, isto é, na escolha de clientes que possuam avaliações similares de produtos, quanto baseadas em itens, na escolha de produtos avaliados similarmente (10).

Mais formalmente, quando a filtragem colaborativa é baseada em usuários, a utilidade ℓ_{ui} de um item i para um usuário u é estimada com base nas utilidades $\ell_{v_k^u i}$ dos usuários $v_k^u \in \mathcal{U}$ que são "similares" ao usuário u. De maneira análoga, quando baseada em itens, a utilidade ℓ_{ui} é prevista com base nas utilidades $\ell_{uj_k^u}$, dado itens $j_k^u \in \mathcal{I}$ que são "similares" aos itens i.

Na prática, o cálculo das recomendações para sistemas colaborativos é feito a partir da matriz de avaliações $\mathbf{R}$. Isso pode ser exemplificado pela Tabela 2, que possui avaliações de 1 a 5, sendo $M = 2$. Em um sistema usuário-usuário, o cliente u_1 receberia recomendação do item i_4, pois para os itens i_2 e i_3 suas avaliações foram similares às do cliente u_2. Já para um sistema item-item, o usuário u_3 receberia recomendação do item i_3, pois este tem avaliações similares às do item i_2, avaliado positivamente pelo usuário u_3.

Tabela 2 – Avaliações r_{ui}

	i_1	i_2	i_3	i_4
u_1	-	4	3	-
u_2	-	4	3	5
u_3	2	5	-	1

Por fim, as recomendações híbridas combinam aspectos tanto da filtragem colaborativa (baseada em usuários ou em itens) quanto da filtragem baseada em conteúdo, com o objetivo de atingir uma melhor recomendação ou de superar problemas recorrentes nas técnicas individuais, como a esparsidade (*sparsity*) dos dados ou o *cold start* (11).

3.2 Estado da arte das soluções

Do ponto de vista do estado da arte das soluções, as variáveis de interesse estão ligadas do número de usuários no sistema, ao número de itens, à medida de qualidade da recomendação e ao custo computacional (12).

No que se refere à dependência do número de usuários, a filtragem colaborativa baseada em usuários é extremamente efetiva para um baixo número de usuários. A filtragem colaborativa a base de itens é consideravelmente pior para um baixo número de usuários, mas supera todos os outros métodos baseados em memória conforme o número de clientes aumenta.

A dependência do número de itens é, de certa forma, oposta à de usuários: a filtragem colaborativa baseada em itens é extremamente efetiva para poucos itens, enquanto aquela baseada em usuários supera todos os outros métodos baseados em memória para grandes quantidades de itens.

Com relação à medida de qualidade, avaliada a partir da acurácia dos dados, a filtragem baseada em usuários e a baseada em itens mostram uma dependência semelhante. Na análise de menor erro quadrático médio entre o item sugerido e o item efetivamente comprado, todos os métodos de recomendação variam não-linearmente com o número de usuários, itens e acurácia, e de modo geral há um compromisso (*trade-off*) entre a esparsidade (*sparsity*) dos dados e o tempo de processamento.

3.3 Desafios científicos e tecnológicos

Um dos maiores desafios tecnológicos dos sistemas de recomendação é, atualmente, o da escalabilidade (7). O sistema de recomendação deve ser flexível no sentido de poder operar igualmente bem tanto em pequenas quanto em grandes bases de dados, que podem chegar até centenas de milhões de clientes (13) e de produtos (14). Isso significa que as recomendações devem ser suficientemente rápidas e ainda assim prover sugestões valiosas aos consumidores.

Um problema muito comum nos sistemas de recomendação é o do *cold start*, que atinge principalmente os sistemas de filtragem colaborativa, grandemente dependentes da matriz de avaliações $\mathbf{R}$. Quando itens ou usuários são inicialmente introduzidos no sistema, existe pouca ou nenhuma informação sobre eles. O sistema é incapaz de realizar inferências sobre quais itens recomendar ao novo usuário ou sobre quais produtos são similares ao novo item. Na Tabela 3, por exemplo, o item i_{100} não possui nenhuma avaliação, e nunca seria recomendado em sistemas puramente baseados em itens. Analogamente, o usuário u_3 também não teria nenhuma sugestão de itens para sistemas puramente baseados em usuários.

Tabela 3 – Avaliações r_{ui}

	i_1	i_2	$\cdots$	i_{100}
u_1	5	4	$\cdots$	-
u_2	-	2	$\cdots$	-
u_3	-	-	$\cdots$	-

Também é uma grande dificuldade dos algoritmos baseados em filtragem colaborativa a esparsidade dos dados. Como a maioria dos clientes interage com uma pequena quantidade de itens, a matriz de avaliações tem em geral menos de 1% dos valores preenchidos, e o sistema deve prever os outros valores (15).

Outro desafio científico é referente à diversidade das recomendações realizadas, também chamado de excesso de especialização (*over-specialization*) (4). Ao mesmo tempo que o sistema deve apresentar itens similares ao que o usuário está procurando, ele também deve sugerir itens que o usuário desconheça ou que nem saiba que poderiam interessá-lo. Esse problema afeta principalmente os sistemas baseados em conteúdo, pois itens com características similares tendem a ser sempre recomendados. Na Tabela 4, o item i_2 seria sugerido para um usuário que tenha avaliado i_1, apesar de esse não apresentar nenhuma característica diferente do item previamente comprado. Para contornar essa dificuldade, costuma-se introduzir elementos de aleatoriedade na recomendação, por exemplo a partir de algoritmos genéticos (6).

Tabela 4 – Atributos a_{if}

	f_1	f_2	f_3
i_1	1	50	0.8
i_2	1	50	0.8
i_3	0	75	0.3

Além desse desafio, existe também o da análise rasa do conteúdo (*shallow content analysis*). O sistema, ao avaliar a característica dos itens, não consegue extrair importantes aspectos para o usuário caso eles não estejam explicitamente descritos na categorização do banco de dados. Isso pode ocorrer, por exemplo, com fatores externos ao produto que influenciem na compra do usuário, como em datas comemorativas, em compras induzidas por propaganda, em compras "impulsivas", etc. Se um usuário comprou um arranjo de flores no dia das mães, não é necessariamente verdade que ele se interessa por flores. Da mesma maneira, sistemas que ignorem a sazonalidade de certos produtos não recomendariam arranjos de flores para clientes que não tenham comprado itens parecidos, mesmo no dia das mães.

Por fim, um desafio científico que este trabalho enfrentará é a execução de um sistema híbrido do ponto de vista de efemeridade e persistência, ao construir um modelo de

recomendação que integre as preferências de curto e longo termo dos usuários (2). A análise dos dados de compras anteriores, bem como de dados demográficos, deverá portanto ser incorporada à análise de característica dos produtos, a fim de enriquecer a acurácia do sistema (7).

Esse tópico de pesquisa inclui ainda diversos desafios científicos e tecnológicos que não serão tratados no nosso projeto, tais como a preservação da privacidade dos usuários, a criação de modelos de recomendação inter-domínios, o desenvolvimento de sistemas descentralizados operando em redes computacionais distribuídas, a otimização de sistemas para sequências de recomendações, a otimização de sistemas para dispositivos móveis e outros. Um sistema de recomendação inteligente também deveria prever quando enviar uma determinada recomendação, e não agir apenas mediante requisição dos clientes (16).

3.4 Soluções propostas

Este Trabalho de Conclusão de Curso aborda três propostas de solução para o problema da recomendação, sendo duas delas retiradas de referências bibliográficas (5, 17), e uma outra apresentada pela dupla. O objetivo é realizar uma análise comparativa entre cada um dos métodos e estabelecer diretrizes para sua aplicação em e-commerces. Os algoritmos propostos estão descritos com maior detalhe no Capítulo 5.

Todas as soluções são algoritmos híbridos, por utilizarem na recomendação tanto a matriz de avaliações $\mathbf{R}$ quanto a matriz de atributos $\mathbf{A}$. Optou-se por dar importância aos algoritmos híbridos em razão de os e-commerces estruturarem seus bancos de dados em torno da descrição dos itens à venda. De modo geral, as tabelas de itens possuem dezenas de atributos, dependendo do ramo de negócios da loja, e pouco detalhe é dado à interação entre o grupo de usuários e itens. A tabela de histórico de compras se limita a informações como data e método de pagamento, e detém pouca informação adicional que possa ser utilizada na recomendação de produtos. Dessa forma supusemos que métodos puramente colaborativos, fundamentados na avaliação dos itens por parte dos usuários, teriam pior desempenho que métodos baseados em conteúdo, que exploram as características dos itens na recomendação.

A solução *FW* determina a similaridade de dois itens a partir de medidas de distância para cada um dos atributos dos itens, ponderadas por pesos determinados na regressão linear de uma equação descrita pelo interesse dos usuários em cada *feature*.

O método *UP* parte do princípio que os usuários estão interessados nos atributos dos itens, e traça correlações entre esses dois elementos para obter pesos que servirão de base para o cálculo da similaridade inter-usuários, utilizada na recomendação pelo método da vizinhança (*nearest neighbors*).

A variante *UI*, elaborada pela dupla, recomenda o melhor item a partir das matrizes de correlação usuário-atributo e atributo-item, a fim de obter a matriz de correlação usuário-item. Espera-se que essa solução tenha desempenho similar ao método de base *UP*, pois ambos buscam explorar as características dos itens para determinar a preferência do usuário.

4 Requisitos

A partir dos objetivos deste Trabalho de Conclusão de Curso, é possível extrair os requisitos funcionais do sistema de recomendação. Esses requisitos ditam principalmente sobre a escalabilidade e o desempenho das recomendações do sistema.

Como as sugestões serão calculadas com antecedência, não há necessidade para uma elevada taxa de recomendações por período de tempo (*throughput*). Deseja-se contudo que o sistema possa gerar todas as recomendações para um banco de dados de cem mil clientes em uma hora, isto é, que tenha *throughput* mínimo de 28 recomendação por segundo. Os sistemas de recomendação tradicionais possuem *throughput* de cerca de 500 recomendações por segundo, mas operam em servidores dedicados de maior potência computacional (18).

A fim de poder estabelecer uma base comparativa entre o sistema proposto *UI* e os sistemas de referência *FW* e *UP*, serão utilizados os mesmos indicadores de desempenho dos artigos-base: precisão, abrangência e medida F_1 (5, 17). Precisão é a porcentagem de casos corretamente preditos em relação ao tamanho da lista de recomendações. Abrangência é a razão entre o número de itens corretamente preditos e daqueles que foram efetivamente avaliados pelo usuário. A medida F_1, por sua vez, é a média harmônica entre precisão e abrangência.

Todas essas métricas são dependentes dos diversos parâmetros do problema, como do tamanho da lista de recomendações N, da quantidade de vizinhos mais próximos k, e principalmente do banco de dados de teste. Como os artigos de referência não os disponibilizaram integralmente, serão estimados os valores de precisão, abrangência e medida F_1 para o banco de dados da dupla.

Espera-se que a precisão, abrangência e consequentemente a medida F_1 sejam maiores que 20%. Esses valores foram escolhidos por serem superiores aos de algoritmos puramente baseados em conteúdo ou em filtragem colaborativa (5, 17). Na prática, o resultado mais importante é a comparação entre os três métodos para um banco de dados de referência.

Neste trabalho o *benchmark* é feito por meio de dois bancos amplamente utilizados na comunidade científica de Sistemas de Recomendação. O primeiro, denominado MovieLens 100k, é composto de 100 000 avaliações (valores inteiros de 1 a 5) de 943 usuários para 1682 filmes (19). Além disso, cada usuário (idade, sexo, profissão, logradouro) avaliou pelo menos 20 filmes (gênero, ano de publicação). O segundo banco de dados é extraído do Internet Movie Database (IMDB), e possui 28 819 filmes. Esse banco está presente na biblioteca `ggplot2` da linguagem de programação R (20).

Os requisitos funcionais são suportados por requisitos não-funcionais, e estes são determinados pelas restrições sobre o projeto ou execução, tais como desenvolvimento e confiabilidade.

O sistema de recomendação deverá poder ser utilizado por qualquer e-commerce que disponha de um banco de dados de clientes, produtos e histórico de compras, desde que o formato de entrada, a ser especificado no Capítulo 7, seja seguido.

Além disso o sistema deverá ser desenvolvido em tecnologias abertas (*open source*) que tenham um alto número de colaboradores, como o sistema de gestão de banco de dados MySQL ou a linguagem de programação estatística R, a fim de torná-lo reutilizável por alunos ou e-commerces interessados.

Por fim, o sistema de recomendação deverá ser escalável e flexível no sentido de poder operar igualmente bem tanto em pequenas quanto em grandes bases de dados.

Apesar serem importantes parâmetros de um sistema de recomendação, a taxa de recomendações por período de tempo e a escalabilidade estão intimamente relacionados ao orçamento do projeto. Pode-se obter virtualmente qualquer *throughput* desejado, contanto que haja investimento equivalente em infra-estrutura computacional. O mesmo não é válido para os parâmetros de qualidade da recomendação, que dependem tão somente dos algoritmos de sugestão. Neste trabalho, assumimos que o sistema de operará em microcomputadores pessoais, e por isso o requisito funcional *throughput* se faz necessário.

5 Síntese de Soluções

5.1 Algoritmo baseado na ponderação de atributos (FW)

O primeiro algoritmo que utilizaremos no sistema de recomendação, adaptado da Referência 5 e denominado ponderação de atributos, *feature weighting* ou *FW*, trata-se de um híbrido entre filtragem colaborativa e filtragem baseada em conteúdo. A partir da regressão linear de dados de uma rede social (*Internet Movie Database, IMDB*), extraem-se os pesos que determinam a importância de cada atributo dos itens, e é onde ocorre a filtragem colaborativa dos usuários. Após obtenção dos pesos, realiza-se a filtragem baseada em conteúdo para determinar os itens com maior similaridade, que são finalmente recomendados.

Na filtragem baseada em conteúdo, "cada item é representado por um vetor de atributos ou *features*". A similaridade s_{ij} entre dois itens i e j é dada pela média ponderada das distâncias entre as *features* dos itens:

$$s_{ij} = \sum_{f} w_f \left(1 - d_{fij}\right) \tag{5.1}$$

As distâncias entre os atributos d_f são determinadas conforme o tipo de dado avaliado e seu domínio, normalizadas no intervalo $[0, 1]$.

Para atributos literais, como categoria, marca, cor, etc., uma possível medida de distância é o delta de Kronecker descrito em 5.2. A similaridade entre as cores "azul" e "vermelho" é, nesse caso, 0, e sua distância é 1. O valor da distância é nulo se e somente se os atributos são idênticos.

Para atributos pertencentes a uma coleção finita de itens, tais como os atores participantes de um filme, é possível estabelecer a similaridade entre dois conjuntos a partir do índice Jaccard, descrito em 5.3. Neste caso, a similaridade entre os conjuntos {Al Pacino, Tom Hanks} e {Tom Hanks, Marlon Brando} é 1/3, e a sua distância é 2/3.

$$\delta_{mn} = \begin{cases} 1, & \text{se } m = n \\ 0, & \text{se } m \neq n \end{cases} \tag{5.2}$$

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{5.3}$$

Vale considerar a correlação entre atributos no cálculo das distâncias: a similaridade de duas marcas de calçado, por exemplo, é maior que a de duas marcas de produtos de

categorias diferentes, mesmo que as marcas sejam distintas nos dois casos. Em uma primeira análise, todavia, utilizaremos para a maior parte das *features* as medidas de distância do delta de Kronecker 5.4 e do índice Jaccard 5.5. Isso significa que se os atributos de dois itens são idênticos, a distância é nula e portanto a similaridade é máxima. O sumário de algumas medidas de distância que podem ser utilizadas estão na Tabela 5.

$$\begin{aligned} d_{fij} &= 1 - \delta_{ij}^{f} \\ &= 1 - \delta_{a_{if}a_{jf}} \end{aligned} \tag{5.4}$$

$$\begin{aligned} d_{fij} &= 1 - J^{f}(i,j) \\ &= 1 - J(a_{if}, a_{jf}) \end{aligned} \tag{5.5}$$

Tabela 5 – Medidas de distância entre alguns atributos

Atributo f	**Domínio** F	**Distância** d_f
Marca	Literal	$1 - \delta_{ij}^{f}$
Esporte	Literal	$1 - \delta_{ij}^{f}$
Gênero	Literal	$1 - \delta_{ij}^{f}$
Categoria	Conjunto Literal	$1 - J^{f}(i,j)$
Preço	$\mathbb{R}$	$\frac{\lvert a_{if} - a_{jf} \rvert}{\max_{i,j} \lvert a_{if} - a_{jf} \rvert}$
Data	$\mathbb{R}$ milissegundos a partir de *epoch* (21)	$\frac{\lvert a_{if} - a_{jf} \rvert}{\max_{i,j} \lvert a_{if} - a_{jf} \rvert}$

Os pesos w_f são a priori desconhecidos. A Referência 5 os determina a partir de uma regressão linear do tipo 5.6, onde e_{ij} é o número de usuários que se interessam tanto por i quanto por j. Esses valores permitem determinar "o julgamento humano de similaridade entre itens", e pode ser calculado a partir da matriz de avaliações, conforme a equação 5.7. O operador booleano b_M, descrito pela Equação 5.8, nada mais é que uma ferramenta matemática para se poder extrair o número de usuários que avaliaram *positivamente* tanto i quanto j a partir de $\mathbf{R}$.

$$e_{ij} = w_0 + \sum_{f} w_f \left(1 - d_{fij}\right) \tag{5.6}$$

$$e_{ij} = \sum_{u} \mathrm{b_M}\left(r_{ui}\ r_{uj}\right) \tag{5.7}$$

$$\mathrm{b}_M\left(x\right) = \begin{cases} 1, & \text{se } x > M \\ 0, & \text{se } x \leq M \end{cases} \tag{5.8}$$

Desta forma, os pesos w_f são determinados a partir resolução do sistema de equações lineares 5.9. Apenas os pesos positivos e com valor absoluto expressivo (maior que um piso arbitrariamente escolhido a posteriori) são utilizados na recomendação.

$$w_0 + \sum_f w_f \,(1 - d_{fij}) = \sum_u \mathrm{b}_0 \,(r_{ui}\ r_{uj}),\ \forall i \neq j \tag{5.9}$$

Calcula-se a matriz de similaridade $\mathbf{S}$ pela equação 5.1 e recomendam-se os itens similares àqueles já comprados, segundo 5.10.

$$\hat{\imath}_u = \underset{i \in \{i \ \mid \ r_{ui} > 0\}, j}{\arg\max}\ s_{ij} \tag{5.10}$$

5.2 Algoritmo baseado no perfil de usuários (UP)

O segundo algoritmo, adaptado da Referência 17, é um hibrido entre filtragem colaborativa e filtragem baseada em conteúdo. Os atributos dos itens são ponderados no cálculo de similaridade, com pesos extraídos de um modelo de perfil de usuários, denominado *user profile* ou *UP*. Esse perfil leva em consideração o interesse dos usuários por *features*, indiretamente calculado a partir de seu interesse pelos itens.

Para se determinar a relevância de f para u, deve-se levar em conta não somente a frequência com a qual uma característica aparece, mas também o fato de algumas características estarem contidas na maioria dos itens. Determina-se, então, os pesos w_{uf}, que mostram a relevância de f para u, a partir da medida estatística TF-IDF (*term frequency–inverse document frequency*), presente em formulações de recuperação de informação e mineração de dados (Equação 5.13.

Em nosso caso, TF ou *feature frequency* é a "similaridade intra-usuários", igual ao número de vezes em que a *feature* f aparece no perfil do usuário u (Equação 5.11). Se o usuário avaliou *positivamente* algum item r_{ui}, tal que r_{ui} é superior a um valor mínimo M, considera-se que u tem interesse TF_{uf} nos atributos f dos itens i, representados por a_{if}.

$$\mathrm{TF}_{uf} = \sum_i \mathrm{b}_M \,(r_{ui}\ a_{if}) \tag{5.11}$$

O termo IDF ou *inverse user frequency* é a "dissimilaridade inter-usuários", relacionada com o inverso da frequência de um atributo f dentro de todos os usuários (Equação 5.12).

$$\mathrm{IDF}_f = \log\left(\frac{\mid \mathcal{U} \mid}{\sum_u \mathrm{b}_0 \,(\mathrm{TF}_{uf})}\right) \tag{5.12}$$

Os pesos w_{uf}, obtidos na TF-IDF 5.13, são utilizados para calcular a similaridade s_{uv} entre dois usuários u e v, conforme as Equações 5.14 e 5.15.

$$w_{uf} = \mathrm{TF}_{uf}\ \mathrm{IDF}_f \tag{5.13}$$

$$s_{uv} = \frac{\sum\limits_{f \in \mathcal{F}_{uv}} w_{uf}\ w_{vf}}{\sqrt{\sum\limits_{f \in \mathcal{F}_{uv}} w_{uf}^2}\sqrt{\sum\limits_{f \in \mathcal{F}_{uv}} w_{vf}^2}} \tag{5.14}$$

$$\begin{aligned} \mathcal{F}_{uv} &= \mathcal{F}_u \cap \mathcal{F}_v \\ \mathcal{F}_u &= \{f \in \mathcal{F} \mid t_{uf} > 0\} \end{aligned} \tag{5.15}$$

Dispondo-se de $\mathbf{S}$, selecionam-se os k vizinhos mais próximos v_k^u com maior similaridade s_{uv}. Posteriormente, determina-se o conjunto $\mathcal{I}_{v_k^u} = \left\{i \mid r_{v_k^u i} > M\right\}$ de itens i avaliados positivamente por v_k^u. Em 5.16 avalia-se a frequência total f_{uf} dos atributos f para os itens de $\mathcal{I}_{v_k^u}$.

$$\mathrm{f}_{uf} = \sum_{i \in \mathcal{I}_{v_k^u}} \mathrm{b}_0\left(a_{if}\right) \tag{5.16}$$

Por fim, a partir da equação 5.17 calcula-se o peso ω_{ui} de cada item e gera-se a lista dos *top-N* produtos a serem recomendados para o usuário u, conforme 5.18.

$$\omega_{ui} = \sum_f a_{if}\ \mathrm{f}_{uf} \tag{5.17}$$

$$\hat{\imath}_u = \underset{i \in \{i \mid r_{ui} > 0\}}{\arg\max}\ \omega_{ui} \tag{5.18}$$

5.3 Algoritmo baseado na correlação usuário-item (UI)

Este método se trata de uma variante da solução *UP*, e também está embasado no cálculo da preferência do usuário por *features*, medida através do seu interesse pelos itens. O algoritmo *UI* utiliza as matrizes de correlação ponderada entre usuários e atributos $\mathbf{W}$ e a matriz de atributos dos itens $\mathbf{A}$ no cálculo da correlação usuário-item.

A lista dos N produtos a serem recomendados decorre portanto do cálculo de ω_{ui} (Equação 5.19) e da escolha dos itens que maximizem essa variável para cada usuário (Equação 5.18).

$$\omega_{ui} = \sum_f w_{uf}\ a_{if} \tag{5.19}$$

Ao passo que o método *UP* recomenda itens a partir dos k vizinhos mais próximos, o algoritmo *UI* busca os itens com *features* mais similares aos atributos pelos quais u se interessa, diretamente através da matriz de atributos. Espera-se que esse tipo de recomendação forneça sugestões de qualidade similar ao algoritmo original, pois os dois tem a mesma fundamentação inicial.

6 Avaliação de Desempenho

De modo geral os sistemas de recomendação tem o objetivo de apresentar ao usuário itens pelos quais ele possa se interessar. O desempenho de um sistema de recomendação se mede, portanto, na qualidade com a qual ele executa essa tarefa.

Essa qualidade pode ser medida de diferentes maneiras, tal como pela medida de distância entre os produtos recomendados $\hat{\mathbf{i}}$ e aqueles que seriam efetivamente comprados $\mathbf{i}$ pelo cliente em uma validação cruzada (*cross validation*). Outras medidas de predição também podem ser utilizadas, a exemplo de trabalhos de recuperação de informação, tais como acurácia (*accuracy*), especificidade (*specificity*), precisão (*precision*), abrangência (*recall*), medida F_1 (F_1-*score*), e outras.

No nosso Trabalho de Conclusão de Curso, serão utilizados precisão, abrangência, e medida F_1. Essas medidas foram escolhidas a fim de se poder estabelecer uma base comparativa com os textos de referência, que também as utilizam. Elas estão sumarizadas na Tabela 6. As quantidades VP, FP, VN e FN significam o número de verdadeiro e falso positivos e o número de verdadeiro e falso negativos.

Tabela 6 – Avaliação de sistemas de predição

Medida	**Fórmula**	**Significado**
Precisão	$\frac{VP}{VP+FP}$	Porcentagem de casos positivos corretamente preditos.
Abrangência	$\frac{VP}{VP+FN}$	Porcentagem de casos positivos sobre aqueles que foram marcados como positivos.
F_1	$2 \cdot \frac{\text{Precisão} \cdot \text{Abrangência}}{\text{Precisão} + \text{Abrangência}}$	Média harmônica entre precisão e abrangência.

Por fim, avaliaremos o desempenho do sistema mediante a mudança nas variáveis de importância do problema, como por exemplo na quantidade de atributos utilizados na recomendação. O tempo de execução também será avaliado em função do algoritmo utilizado e do tamanho do banco de dados.

7 Resultados

Até o presente momento, os resultados deste Trabalho de Conclusão de Curso concentram-se na definição de necessidades e de parâmetros de sucesso e elaboração de possíveis soluções.

Visto que a primeira etapa de um sistema de recomendação é a extração de informações, definimos que a aquisição de dados será feita a partir de uma base genérica, que deverá alimentar o sistema por meio de arquivos de texto com valores separados por vírgulas (`.csv`).

A fim de facilitar o pré-processamento dos dados, exigem-se três arquivos, cada um com uma tabela de itens e seus atributos $\mathbf{A}$, clientes e suas características $\mathbf{B}$ e histórico de compras ou avaliações $\mathbf{R}$. Caso existam outras informações no banco de dados, o sistema deverá ser alterado para levar em conta o processamento dos arquivos suplementares.

$$\mathbf{A} = \begin{bmatrix} a_{i_1 f_1} & a_{i_1 f_2} & a_{i_1 f_3} & \dots \\ a_{i_2 f_1} & a_{i_2 f_2} & a_{i_2 f_3} & \dots \\ a_{i_3 f_1} & a_{i_3 f_2} & a_{i_3 f_3} & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{7.1}$$

$$\mathbf{B} = \begin{bmatrix} b_{u_1 c_1} & b_{u_1 c_2} & b_{u_1 c_3} & \dots \\ b_{u_2 c_1} & b_{u_2 c_2} & b_{u_2 c_3} & \dots \\ b_{u_3 c_1} & b_{u_3 c_2} & b_{u_3 c_3} & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{7.2}$$

$$\mathbf{R} = \begin{bmatrix} r_{u_1 i_1} & r_{u_1 i_2} & r_{u_1 i_3} & \dots \\ r_{u_2 i_1} & r_{u_2 i_2} & r_{u_2 i_3} & \dots \\ r_{u_3 i_1} & r_{u_3 i_2} & r_{u_3 i_3} & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{7.3}$$

Em alguns bancos de dados relacionais, a tabela de histórico também contém outras informações adicionais θ, tais como método de pagamento, data da compra, data de entrega, etc., e é denominada $\mathbf{H}$.

$$\mathbf{H} = \begin{bmatrix} r_{u_1 i_1} & \theta_{h_1 1} & \theta_{h_1 2} & \dots \\ r_{u_1 i_2} & \theta_{h_2 1} & \theta_{h_2 2} & \dots \\ r_{ui} & \theta_{h1} & \theta_{h2} & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{7.4}$$

Uma vez determinada a forma de entrada de dados, definiu-se o conjunto de dados a serem utilizados. O primeiro conjunto de dados abertos é proveniente do website de recomendações de filmes MovieLens (<http://movielens.umn.edu>). Nessa base de dados, o catálogo de filme faz o papel de catálogo de produtos pelos quais os usuários possam se interessar, e o histórico de compras se refere à avaliação dos filmes feita por cada usuário. Outros conjuntos de dados abertos, a serem definidos posteriormente, também poderão ser explorados pela dupla, tais como os dados de classificação de músicas do serviço Yahoo! Music (<http://webscope.sandbox.yahoo.com>).

Outra base que poderemos utilizar é a de dados anônimos de lojas de comércio online. Já entramos em contato com um e-commerce de vendas de passagens de ônibus interessado em sistemas de recomendação, que forneceria sua base para nossa pesquisa. Ao longo das últimas semanas, estabelecemos os termos do contrato de confidencialidade e explicamos como seria o desenrolar do trabalho. Esperamos obter o banco de dados antes das férias escolares, a fim de aplicar os algoritmos de recomendação também nesse caso específico. Caso a parceria se concretize, será possível avaliar o desempenho do sistema de recomendação sob aspectos como aumento no número de compras, receita adicional gerada, comparação entre os métodos anteriores utilizados pela loja e o nosso sistema, etc.

Mesmo sem ter o banco de dados em mãos, já nos foi explicado sua estrutura, para que possamos adaptar a entrada e saída de dados genérica descrita acima para esse caso específico. Ele é dividido em seis diferentes tabelas MySQL mostradas abaixo. Os relacionamentos entre elas e seu conteúdo estão demonstrados na Figura 1.

1. Ordem (pedido)
2. Itens da ordem
3. Rotas
4. Lugares
5. Pagamentos de ordens
6. Endereço de cobrança

Outro tópico discutido na reunião foi a importância da data de recomendação. O sistema deve não só prever qual será a próxima viagem do usuário, mas também quanto tempo antes da viagem devemos oferecê-la ao cliente. A loja de passagens de ônibus informou que, após estudo interno, descobriu-se que 60% das passagens são compradas até três dias antes da data de embarque. Isso mostra que uma recomendação que não leve em conta fatores como data de viagem ou sazonalidade (o número de viagens em datas comemorativas é em geral cinco vezes maior que a média diária) terá um mau desempenho.

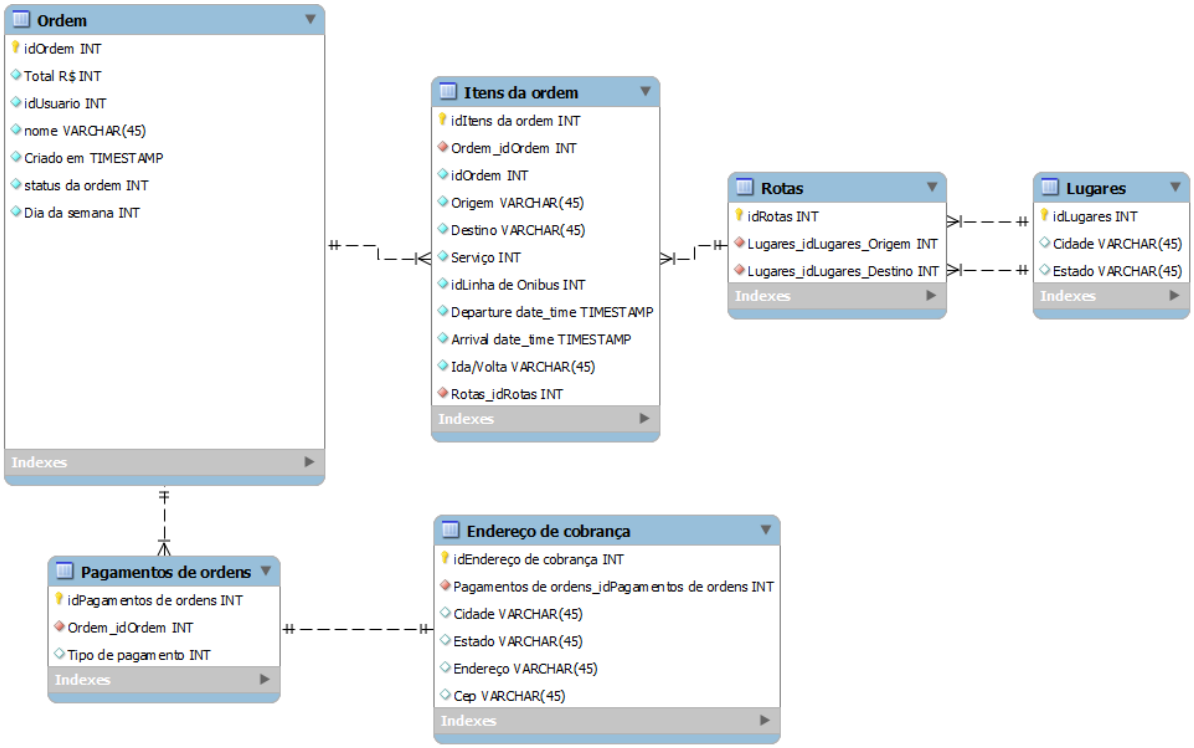

Figura 1 – Relacionamento entre as tabelas do banco de dados de passagens de ônibus

Por fim, definimos que para qualquer que seja a aplicação, os resultados das recomendações serão entregues por meio de um arquivo `.csv`. Este conterá= o identificador de cada usuário com as recomendações de produtos, assim como o valor numérico associado à recomendação. Esse resultado é o mais importante do ponto de vista do e-commerce, que o utilizará como estratégia de marketing na sugestão de produtos.

Referências

1 RICCI, L. R. F.; SHAPIRA, B. Introduction to recommender systems handbook. In: *Recommender Systems Handbook*. [S.l.]: Springer, 2011. p. 1–35. Citado na página 4.

2 SCHAFER, J. B.; KONSTAN, J.; RIEDL, J. Recommender systems in e-commerce. In: ACM. *Proceedings of the 1st ACM conference on Electronic commerce*. [S.l.], 1999. p. 158–166. Citado 2 vezes nas páginas 4 e 11.

3 SARWAR, B. et al. Analysis of recommendation algorithms for e-commerce. In: ACM. *Proceedings of the 2nd ACM conference on Electronic commerce*. [S.l.], 2000. p. 158–167. Citado na página 5.

4 ADOMAVICIUS, G.; TUZHILIN, A. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *Knowledge and Data Engineering, IEEE Transactions on*, IEEE, v. 17, n. 6, p. 734–749, 2005. Citado 3 vezes nas páginas 6, 7 e 10.

5 SYMEONIDIS, P.; NANOPOULOS, A.; MANOLOPOULOS, Y. Feature-weighted user model for recommender systems. In: *User Modeling 2007*. [S.l.]: Springer, 2007. p. 97–106. Citado 5 vezes nas páginas 6, 11, 13, 15 e 16.

6 BALABANOVIC, M.; SHOHAM, Y. Fab: Content-based, collaborative recommendation. *Communications of the ACM*, v. 40, p. 66–72, 1997. Citado 2 vezes nas páginas 7 e 10.

7 WEI, K.; HUANG, J.; FU, S. A survey of e-commerce recommender systems. In: IEEE. *Service Systems and Service Management, 2007 International Conference on*. [S.l.], 2007. p. 1–5. Citado 3 vezes nas páginas 7, 9 e 11.

8 SCHAFER, J. B.; KONSTAN, J. A.; RIEDL, J. E-commerce recommendation applications. *Data Mining and Knowledge Discovery*, v. 5, p. 115–153, 2001. Citado na página 7.

9 LOPS, P.; GEMMIS, M. de; SEMERARO, G. Content-based recommender systems: State of the art and trends. In: *Recommender Systems Handbook*. [S.l.]: Springer, 2011. p. 73–105. Citado na página 7.

10 LINDEN, G.; SMITH, B.; YORK, J. Amazon.com recommendations: Item-to-item collaborative filtering. *Internet Computing, IEEE*, IEEE, v. 7, n. 1, p. 76–80, 2003. Citado na página 8.

11 BURKE, R. Hybrid web recommender systems. In: *The adaptive web*. [S.l.]: Springer, 2007. p. 377–408. Citado na página 8.

12 LEE, J.; SUN, M.; LEBANON, G. A comparative study of collaborative filtering algorithms. *arXiv preprint arXiv:1205.3193*, 2012. Citado na página 9.

13 TUTOL, L. *Amazon Launches 'Login and Pay with Amazon' for a Seamless Buying Experience*. 2013. Disponível em: <http://services.amazon.com/post/Tx2A98P3EKP62O2/Amazon-Launches-Login-and-Pay-with-Amazon-for-a-Seamless-Buying-Experience>. Citado na página 9.

14 PALLADINO, V. *Amazon sold 426 items per second in run-up to Christmas*. 2013. Disponível em: <http://www.theverge.com/2013/12/26/5245008/amazon-sees-prime-spike-in-2013-holiday-season>. Citado na página 9.

15 FENNELL, J. Collaborative filtering on sparse rating data for yelp. com. 2009. Citado na página 10.

16 LOPS, P.; GEMMIS, M. de; SEMERARO, G. In: *Recommender Systems Handbook*. [S.l.]: Springer, 2011. Citado na página 11.

17 DEBNATH, S.; GANGULY, N.; MITRA, P. Feature weighting in content based recommendation system using social network analysis. In: ACM. *Proceedings of the 17th international conference on World Wide Web*. [S.l.], 2008. p. 1041–1042. Citado 3 vezes nas páginas 11, 13 e 17.

18 SARWAR, B. et al. Item-based collaborative filtering recommendation algorithms. In: ACM. *Proceedings of the 10th international conference on World Wide Web*. [S.l.], 2001. p. 285–295. Citado na página 13.

19 MOVIELENS. *MovieLens 100k Dataset*. 1998. Disponível em: <http://files.grouplens.org/datasets/movielens/ml-100k-README.txt>. Citado na página 13.

20 WICKHAM, H. *Movies dataset*. 2006. Disponível em: <http://docs.ggplot2.org/0.9.3.1/movies.html>. Citado na página 13.

21 HOPE, C. *Epoch*. 2014. Disponível em: <http://www.computerhope.com/jargon/e/epoch.htm>. Citado na página 16.