Gabriel Felipe Arakaki

Proposta de algoritmo e desenvolvimento de biblioteca para sistemas de recomendação de produtos de lojas de comércio online

São Paulo, Brasil

22 de maio de 2016

Gabriel Felipe Arakaki

Proposta de algoritmo e desenvolvimento de biblioteca para sistemas de recomendação de produtos de lojas de comércio online

Trabalho de Conclusão de Curso apresentado ao Departamento de Engenharia de Produção da Escola Politécnica da Universidade de São Paulo.

Universidade de São Paulo

Escola Politécnica

Trabalho de Conclusão de Curso

Orientador: Prof. Dr. Davi Noboru Nakano

São Paulo, Brasil

22 de maio de 2016

Gabriel Felipe Arakaki

Proposta de algoritmo e desenvolvimento de biblioteca para sistemas de recomendação de produtos de lojas de comércio online/ A.G.F. Viggiano; F.F.S. Araújo. – São Paulo, Brasil, 22 de maio de 2016

33 p.

Orientador: Prof. Dr. Davi Noboru Nakano

Trabalho de Formatura – Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos.

1. Inteligência artificial. 2. Aprendizado computacional. 3. Comercio eletrônico. 4. Produtos I. Prof. Dr. Fábio Gagliardi Cozman. II. Universidade de São Paulo. Escola Politécnica. III. Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos

Agradecimentos

Agradecemos ao professor Fábio Cozman pela sua orientação e apoio durante todo o projeto. Agradecemos também ao professor Thiago Martins e aos demais orientadores das disciplinas PMR2500 e PMR2550 – Projeto de Conclusão do Curso I e II – por terem nos guiado na elaboração da monografia e por terem sempre exigido trabalhos de alta qualidade. Esse papel é fundamental na valorização do diploma de Engenharia Mecatrônica da Escola Politécnica.

Make things as simple as possible, but not simpler (Albert Einstein)

Resumo

O objetivo deste trabalho é projetar e avaliar o desempenho de um algoritmo de recomendação e uma biblioteca computacional para sistemas de sugestão de produtos de lojas de comércio online. Essa biblioteca tem finalidade de permitir a fácil implementação de um sistema de recomendação genérico para ser utilizado por acadêmicos e *e-commerces* que desejem automatizar o processo de sugestão de itens, tal como em *email marketing*.

A biblioteca foi desenvolvida utilizando-se três diferentes algoritmos de recomendação. O algoritmo baseado na ponderação de atributos, que trata-se de um método híbrido entre filtragem colaborativa e filtragem baseada em conteúdo, onde a partir da regressão linear de dados de uma rede social, extrai os pesos que determinam a importância de cada atributo dos itens. O segundo método, baseado em perfil de usuários, leva em consideração o interesse dos usuários por *features*, indiretamente calculado a partir de seu interesse pelos itens. O terceiro método, baseado na correlação usuário-item, é uma variante do método baseado no perfil de usuários e foi desenvolvido pela dupla. Este método busca os itens com *features* mais similares aos atributos pelos quais o usuário se interessa.

A avaliação comparativa dos métodos mostrou a superioridade do algoritmo de perfil de usuários em quase todos os aspectos, e avaliou os principais parâmetros de influência na qualidade da recomendação. A partir dos resultados empíricos mostrados neste trabalho, é possível estabelecer diretrizes para a elaboração de um sistema de recomendação próprio com base na biblioteca elaborada pela dupla.

Palavras-chaves: Inteligência artificial, Aprendizado computacional, Comercio eletrônico, Produtos.

Abstract

This project's scope is to design and assess a recommender system algorithm and library for e-commerces. The goal of this library is to make the implementation of a generic recommender system simple and easy, so it can be used by the academics and e-commerces willing to automate the suggestion of items, such as in email marketing.

The library was developed using three different recommendation algorithms. The feature weighted is a hybrid method, based on collaborative filtering and content-based filtering, in which a linear regression is calculated from a social-network database, extracting the weights that determine each attribute's importance. The second method, based in user profiles, considers the users' interests in specific features, indirectly calculated by the users' interest in different items. The third method, based in the user-item correlation, is derived from the method based in users' profiles and was developed by the authors. This method searches for them items with the features that are more similar to the attributes that the user has shown interest for.

The comparative assessment of the methods has shown the superiority of the user-profile algorithm in almost all aspects, and has measured the main parameters that affect the recommendation quality. From the empirical results shown in this work, it is possible to establish some guidelines on how to create a recommender system based on the library developed by the authors.

Key-words: Artificial intelligence, Machine learning, e-Commerce, Products.

Lista de tabelas

Lista de ilustrações

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
u, v	Usuários
i, j	Itens
f	Atributos dos itens
$\mathbf{X}_{M\times N},\ \mathbf{X}$	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
$\mathbf{S}, s_{ij}, s_{uv}$	Similaridade entre itens i e j ou entre usuários u e v
$\mathbf{W}, w_{uf}$	Correlação ponderada entre usuário u e atributo f
$\mathbf{\Omega}, \omega_{ui}$	Correlação entre usuário u e item i
$\mathbf{w}, w_f$	Peso do atributo f

Sumário

1 Introdução

1.1 Contextualização do trabalho

Segundo as palavras do professor diretor da Escola Politécnica: “A engenharia deve erradicar a pobreza gerando riqueza, através da geração de empregos e criação de empresas”. Tais palavras foram utilizadas na inauguração do laboratório Ocean em parceria entre o departamento de Engenharia de Produção (PRO) e a grande multinacional Samsung, tendo suas operações dentro da Escola Politécnica. O laboratório é uma parceria de cogestão entre universidade e empresa que tem como principal mérito a geração de valor derivada da sinergia entre as pesquisas da academia e o conhecimento aplicado da indústria.

O Ocean é um dos quatro grandes projetos que o PRO acompanha atualmente, esquematizado pela figura a seguir: (DESENHAR FIGURA QUE SERA COLOCADA AQUI)

Inovalab, que existe e funciona, e é interno ao PRO

Fábrica Didática, que não existe mas seria interno ao PRo

Ocean, que existe e é de cogestão PRO Samsung

NEU, que existe e funciona mas é externo ao PRO

Portanto, é de grande interesse do PRO garantir que o laboratório esteja sendo utilizado da melhor forma possível, portanto é necessário mapear todos os interessados pelo bom funcionamento e traçar um plano de ações e indicadores de forma a garantir o pleno potencial do laboratório.

1.2 Motivação

Como aluno atuante no mercado de tecnologia, o presente autor vivencia na prática a deficiência de comunicação entre o mercado e a comunidade científica-acadêmica, gerando um *gap* entre as demandas das empresas e o conteúdo ensinado nas aulas. Devido ao período de grande evolução exponencial da tecnologia das últimas décadas, é necessário que a comunidade acadêmica e as principais escolas de ensino acompanhem essa evolução oferecendo cursos intra e extra curriculares que acompanhem essas tendências.

Nesse contexto, encontra-se a programação como uma das principais necessidades de ensino, pois esta é a base do funcionamento de grande parte das empresas e *startups* atuais. É muito importante que os futuros gestores dessas empresas entendam o funcionamento dessa operação de "alto nível"para otimizar os processos, fazer uma melhor gestão de

projetos e conseguir identificar possíveis gargalos no sistema.

Desta maneira, o laboratório Ocean se mostra não apenas como uma parceria entre empresa e universidade para P&D e Inovação, mas como uma fonte de *feedbacks* constantes sobre as tendências e necessidades de cada um, que devem ser utilizados para fortalecer a universidade em seus pilares: pesquisa, ensino e extensão.

1.3 Objetivos

O objetivo deste Trabalho de Conclusão de Curso é estabelecer processos e KPIs para avaliar se as necessidades de cada um dos *stakeholders* do Laboratório Ocean está sendo atendida.

2 Revisão Bibliográfica

2.1 Estudos sobre ensino de engenharia

Segundo as palavras do professor diretor da Escola Politécnica: “A engenharia deve erradicar a pobreza gerando riqueza, através da geração de empregos e criação de empresas”. Tais palavras foram utilizadas na inauguração do laboratório Ocean em parceria entre o departamento de Engenharia de Produção (PRO) e a grande multinacional Samsung, tendo suas operações dentro da Escola Politécnica. O laboratório é uma parceria de cogestão entre universidade e empresa que tem como principal mérito a geração de valor derivada da sinergia entre as pesquisas da academia e o conhecimento aplicado da indústria.

O Ocean é um dos quatro grandes projetos que o PRO acompanha atualmente, esquematizado pela figura a seguir: (DESENHAR FIGURA QUE SERA COLOCADA AQUI)

Inovalab, que existe e funciona, e é interno ao PRO

Fábrica Didática, que não existe mas seria interno ao PRo

Ocean, que existe e é de cogestão PRO Samsung

NEU, que existe e funciona mas é externo ao PRO

Portanto, é de grande interesse do PRO garantir que o laboratório esteja sendo utilizado da melhor forma possível, portanto é necessário mapear todos os interessados pelo bom funcionamento e traçar um plano de ações e indicadores de forma a garantir o pleno potencial do laboratório.

3 Metodologia

Por se tratar de um projeto de Engenharia de Software, foi necessário dar ênfase às etapas iterativas de desenvolvimento dos algoritmos na metodologia de projeto deste Trabalho de Conclusão de Curso. Esse processo cíclico, com fases de especificação, desenvolvimento e validação, permitiu obter resultados preliminares e os modificar os algoritmos ao longo da disciplina, ajustando detalhes e melhorando o sistema gradativamente (**??**).

A metodologia de execução do projeto, assim como a de avaliação dos resultados, pode ser consolidada da seguinte maneira:

3.1 Casos de sucesso em parcerias empresa-universidade

3.2 Levantamento das necessidades

3.3 Definição de processos

3.4 Definição de indicadores

Com o crescente número de lojas de comércio online, tornou-se necessário a criação de sistemas que pudessem entender e prever o comportamento de consumidores, a fim de oferecer produtos específicos para cada um deles, aumentando o número de vendas e a satisfação do cliente. Observa-se atualmente que o número de sistemas de recomendação gratuitos, de fácil integração e de código aberto (*open source*) são limitados e não correspondem às necessidades do mercado. Existe, pois, a necessidade da criação de uma biblioteca que possa ser utilizada por e-commerces que desejem estabelecer seu próprio sistema de recomendação ou mesmo por indivíduos interessados na temática da recomendação de itens.

O sucesso do projeto pode ser medido em duas frentes: a primeira, quantitativa, mede a precisão e a abrangência das recomendações. Essas duas medidas devem ser superiores a 20%, e seu significado será melhor detalhado no Capítulo **??**. A segunda, qualitativa, avalia se a biblioteca responde bem aos problemas recorrentes de sistemas de recomendação, tais como a escalabilidade e o excesso de especialização.

Nesta fase do projeto, foram propostas possíveis soluções para o desafio da recomendação. Decidiu-se avaliar dois métodos híbridos do meio acadêmico e um outro elaborado pela dupla.

Após a escolha dos métodos de recomendação, as soluções foram detalhadas matematicamente segundo uma mesma notação, e a estrutura dos algoritmos foi descrita e exemplificada. Neste ponto, escolheu-se também a linguagem de programação R e a forma de entrada e saída de dados, por meio de arquivos `.csv`.

A fim de facilitar o pré-processamento dos dados, estabelecemos que seriam necessários dois arquivos. Um deles deve conter a matriz de atributos **A** e o outro, a matriz de avaliações **R**.

$$\mathbf{A} = \begin{bmatrix} a_{i_1 f_1} & a_{i_1 f_2} & a_{i_1 f_3} & \dots \\ a_{i_2 f_1} & a_{i_2 f_2} & a_{i_2 f_3} & \dots \\ a_{i_3 f_1} & a_{i_3 f_2} & a_{i_3 f_3} & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{3.1}$$

$$\mathbf{R} = \begin{bmatrix} r_{u_1 i_1} & r_{u_1 i_2} & r_{u_1 i_3} & \dots \\ r_{u_2 i_1} & r_{u_2 i_2} & r_{u_2 i_3} & \dots \\ r_{u_3 i_1} & r_{u_3 i_2} & r_{u_3 i_3} & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{3.2}$$

Uma vez determinada a forma de entrada de informações, definiram-se os conjuntos de dados a serem utilizados.

O primeiro conjunto de dados abertos é proveniente do sistema de recomendações de filmes MovieLens (<http://movielens.umn.edu>), e é composto de 100 000 avaliações (valores inteiros de 1 a 5) de 943 usuários para 1682 filmes (**??**). Além disso, cada usuário (idade, sexo, profissão, logradouro) avaliou pelo menos 20 filmes (categoria, ano de publicação). Nessa base de dados, chamada de 100k, o catálogo de filme faz o papel de catálogo de produtos, e o histórico de compras se refere à avaliação dos filmes feita por cada usuário.

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 (**??**).

Na nossa análise, os bancos de dados 100k e IMDB foram utilizados complementarmente. A união desses dois conjuntos deu origem à base 100k-IMDB, composta por 943 usuários, 1682 itens e 25 atributos. Na biblioteca proposta pela dupla, os dados demográficos de usuários não são utilizados.

Ainda na etapa de implementação, confirmamos a validade de cada um dos métodos aplicando-os nas matrizes-referência (Tabelas **??** e **??**).

A fim de realizar um estudo comparativo (*benchmarking*) com os artigos de referência, mantivemos a mesma metodologia de avaliação de qualidade do artigo **??**.

Em particular, implementamos uma validação cruzada considerando $T = 75\%$ do banco de dados como base de treinamento ou aprendizado e os 25% restantes como base de testes. Em seguida, mascaramos $H = 75\%$ das avaliações dos usuários-teste, de modo a medir a qualidade do sistema de recomendação em prever os itens positivamente avaliados. Cerca de uma dezena de parâmetros de interesse foram avaliados para cada um dos métodos (Tabela **??**).

Além disso, não fizemos distinção entre valores não observados (*NA value/NULL value*) e avaliações nulas ($r_{ui} = 0$), pois na maioria dos casos essa simplificação é válida. Esse não é o caso, por exemplo, de sistemas em que o usuário pode deliberadamente dar nota zero para um item.

Sabe-se que a extração de um modelo por meio de uma validação cruzada sobre uma mesma base de dados pode gerar *overfitting* (**??**). Para não cair nesse erro e com foco na reprodutibilidade do trabalho, realizamos todas as amostragens em R utilizando o número 2 como semente aleatória (*state seed*). Dessa forma, os parâmetros calculados para os modelos são sempre os mesmos para qualquer teste de qualidade. Evidentemente, caso se deseje avaliar a performance dos métodos para um outro banco de dados, uma validação cruzada rigorosa deverá ser aplicada.

Como a complexidade dos algoritmos excede o limite dos computadores pessoais da dupla, foi necessário contratar o serviço de computação nas nuvens Amazon Web Services.

Alugamos duas máquinas virtuais do tipo `r3.large`, otimizadas para memória. As máquinas, de especificação 2 vCPU, 15 GB de memória RAM e sistema operacional Amazon Linux AMI release 2014.09 x86_64, baseado em RHEL Fedora, custaram USD 0,175 por hora de uso. Todos os testes foram realizados em poucas horas, e o custo total do projeto foi de apenas R$ 5,70. Uma explicação detalhada da configuração do ambiente de testes se encontra na Seção **??**.

3.5 Avaliação de Desempenho

4 Levantamento das Necessidades

4.1 Visão Geral

4.2 Departamento do PRO

Segundo <——— PLACEHOLDER ———>, o ensino de engenharia pode ser dividido em 3 principais modelos: Acadêmico, Market-Driven e Integrativo. <——— INSERIR TABELA EXPLICATIVA ———->

Embora exista uma tendência das universidades em trabalhar principalmente com o primeiro modelo, ele não deixa de ser um modelo idealizado, portanto cada vez mais é gerado espaço para os outros modelos através de iniciativas da gestão do ensino. Aulas de Empreendedorismo, Marketing e Desenvolvimento de Produto representam muito bem um modelo Market-Driven, ao passo que aulas de Sustentabilidade e Desenvolvimento de Problemas da sociedade representam bem o modelo Integrativo.

No caso do modelo *market-driven*, são abertas as portas para estruturas mais práticas de ensino, e dentro desse contexto observa-se um papel fundamental para o Ocean para o departamento: auxiliar no desenvolvimento de futuros engenheiros para estarem alinhados com as necessidades do mercado.

5 Conclusão

5.1 Discussão

Este Trabalho de Conclusão de Curso cumpriu seus objetivos e antedeu aos requisitos estabelecidos no início do projeto. Foi elaborada uma biblioteca para sistemas de recomendação de produtos de e-commerces e foi estabelecida uma respectiva análise de desempenho dos algoritmos de recomendação.

A avaliação de desempenho dos métodos propostos na biblioteca deste trabalho verificaram resultados já conhecidos no meio acadêmico. Em particular, a dependência entre qualidade de recomendação e tamanho da lista de sugestões se verificou (impacto de N).

Além disso, mostramos que um banco de dados com maior quantidade de avaliações (impacto de H) tem mais relevância que um banco de dados com mais usuários (impacto de T).

Outro resultado do trabalho foi a comprovação do fenômeno de *hidden feedback* (impacto de M). Mesmo que construamos métodos embasados na "avaliação positiva" dos usuários, esse parâmetro pode não ter tanta influência, visto que a maioria das avaliações dos clientes já são de fato positivas.

Também foi verificada a influência da quantidade de vizinhos mais próximos em algoritmos que usam essa metodologia colaborativa (impacto de k). Apesar de influenciar na qualidade da recomendação, esse parâmetro desempenha papel secundário.

Uma outra conclusão importante deste trabalho foi da importância de se escolher *a priori* o conjunto de atributos dos itens (impacto de $\mathcal{F}$). A categorização excessiva dos itens pode ser maléfica para a recomendação, caso as *features* não tenham relevância para os usuários.

Avaliamos também diferentes medidas de distância entre os atributos (impacto de d_{ij}^f). A medida da diferença em valor absoluto foi comparada com outros índices, como o índice Jaccard, para uma lista de gêneros, e verificou-se que a distância L_1 resulta em melhor qualidade de recomendação. Vale ressaltar a importância da escolha das medidas de distância, visto seu impacto no desempenho dos algoritmos, sobretudo nos algoritmos UI e FW.

Por fim, avaliamos também a quantidade de pesos dos atributos no método FW (impacto de W). Vimos que a quantidade de $w_f > 0$ não tem grande impacto na recomendação, visto que o valor dos pesos, em si, já é suficiente para alterar a qualidade das

sugestões.

Apenas o método UP atingiu os requisitos funcionais em termos de precisão e abrangência, para uma combinação específica de parâmetros, como valores pequenos de N. Tanto para esse algoritmo quanto para o método FW, o desempenho é sensivelmente inferior ao relatado nos artigos de referência. O motivo por trás disso é a dissimilaridade entre os bancos de dados. Assim como foi confirmado, o emprego de bases com mais recomendações r_{ui} influencia grandemente na qualidade das recomendações. Para se obter um *benchmarking* mais fiel, seria necessário utilizar o banco de dados dos autores de referência.

Apesar de, por definição, o método UI ser similar ao método UP, seu desempenho foi sensivelmente inferior ao do algoritmo-base. Isso se deve fundamentalmente a dois motivos: o primeiro é devido ao fato de o algoritmo UP possuir uma etapa de correlação entre vizinhos mais próximos, que cumpre com eficácia o papel de selecionar apenas as melhores recomendações. O segundo é devido à má escolha da função que calcula a correlação usuário-item. A simples multiplicação da correlação usuário-atributo w_{uf} pelo valor numérico do atributo do item a_{if} não tem ligação direta com a qualidade de um item para um determinado usuário.

Conforme foi exemplificado no Capítulo **??**, se um usuário gosta filmes de época (elevado w_{uf}), mas apenas de filmes antigos, a multiplicação $w_{uf} \cdot a_{if}$ apontaria que ele se interessa por filmes *atuais* (elevado a_{if}). Faz-se necessário, portanto, substituir a multiplicação direta entre $\mathbf{W}$ e $\mathbf{A}$ por uma expressão que correlacione o valor numérico do atributo com o interesse do usuário pela *feature*, seja $\mathbf{W}$ e $g(\mathbf{A})$, sendo a função g a se determinar.

Mesmo com essa falha de concepção, o algoritmo UP mostrou desempenho similar ao algoritmo FW, superando-o inclusive em tempo de processamento.

5.2 Trabalhos futuros

A extensão desse Trabalho de Conclusão de Curso pode se dar de diversas maneiras, tanto na área acadêmica quanto na área empresarial. Seguindo o atual encaminhamento do projeto, a principal oportunidade do nosso trabalho é a criação de um serviço de um "Sistema de Recomendação nas Nuvens".

Desejamos eliminar as restrições quanto a entrada e saída de dados, de forma que elas fossem completamente arbitrárias. O objetivo é que o usuário possa informar ao sistema como é formado sua base, e que todo o tratamento preliminar seja feito automaticamente.

É possível explorar também a construção de um *driver* que possibilite a conexão entre o sistema de recomendação e um banco de dados SQL, sem que seja necessária a

etapa intermediária de arquivos `csv` para aquisição de dados. Em seguida, é importante elaborar um *website* para o sistema de recomendação e exportar toda a lógica para um servidor dedicado.

Outra melhoria desejada é a melhoria dos métodos e funções, a fim de aprimorar a performance computacional. Dessa forma, o serviço de "sistema de recomendação nas nuvens" estaria completo e poderia ser utilizado por e-commerces reais.

Também seria desejável, para uma avaliação mais completa do trabalho, o emprego dos métodos computacionais em um banco de dados de um e-commerce real. Apesar de termos contatado diversas lojas de comércio online, devido a impedimentos administrativos, não obtivemos sucesso em firmar uma parceria com essas lojas.

No campo acadêmico, há muito espaço para melhorias nos algoritmos de recomendação. Conforme mostrado, a quantidade de atributos, seus pesos e suas medidas de distância tem grande influência na qualidade da recomendação. Seria interessante, portanto, explorar diferentes estratégias de determinação dessas variáveis para todos os métodos. É possível, por exemplo, utilizar algoritmos genéticos ou redes neurais que explorem combinações de parâmetros e pesos a fim de maximizar a precisão e acurácia para uma determinada base.

Além disso, as metodologias de solução de cada um dos sistemas deveriam ser debatidas ao máximo, de modo a explorar casos de uso particulares e a propor mudanças e otimizações. Faz-se necessário responder a perguntas como "O que acontece com itens ou usuários sem nenhuma avaliação?" e "Qual o desempenho dos métodos para outros bancos de dados?".