Sistema de Recomendação Baseado em Confiança para Promover a Colaboração em Redes de Pesquisa Científica

João Pedro R. D. Saldanha[1], **Alexandre Zamberlan**[1]

[1]Ciência da Computação – Universidade Franciscana (UFN)
Rua dos Andradas, 1614 – 97010-032 – Santa Maria – RS – Brasil

{joao.pedro,alexz}@ufn.edu.br

Abstract. *This paper presents a proposal for building a recommender system that promotes collaboration among researchers in a publication network. A trust network is abstracted in which researchers are bounded by joint publications, which represents a mutual trust statement. An architecture is proposed to improve recommendation quality trough profile analysis with metrics for computing trust, using Python with the Pandas and NetworkX libraries.*

Resumo. *Este artigo apresenta a proposta da elaboração de um sistema de recomendações para promover a colaboração entre pesquisadores em uma rede de publicações. É abstraída uma rede de confiança na qual pesquisadores são unidos por publicações em conjunto, que constituem um voto de confiança mútua. Foi proposta uma arquitetura para melhorar as recomendações por meio de análise de perfil, na qual foram implementadas métricas de computação de confiança usando a linguagem Python e as bibliotecas Pandas e NetworkX.*

1. Introdução

O universo da pesquisa científica está em constante expansão, tanto no que diz respeito ao conhecimento produzido quanto ao volume de trabalhos e publicações. Estimativas apontavam um valor em torno de 2.5 milhões de artigos científicos publicados por ano em 2015, com um aumento de 5% ao ano no número de cientistas fazendo publicações [Ware and Mabe 2015]. Pesquisadores não têm o tempo necessário para analisar todos os estudos relacionados à seus próprios trabalhos, mesmo com plataformas como o Lattes, onde tais trabalhos estão compilados. Trata-se do problema da sobrecarga de informação, que tem crescido na medida em que sistemas digitais vem ganhando cada vez mais usuários e conteúdo. Outro problema decorrente do crescimento do número de pesquisadores e trabalhos é que muitas vezes os pesquisadores não conhecem outros pesquisadores da área e acabam por perder a oportunidade de colaborações ou troca de ideias. Logo, faz-se necessário a filtragem da informação que chega ao pesquisador para maximizar sua eficiência e evitar tempo perdido. A automação da tarefa de filtragem é feita através de sistemas de recomendação. Utilizando técnicas de mineração de dados e inteligência artificial pode-se oferecer conteúdo mais relevante, aumentando a eficiência do acadêmico.

A partir do problema da sobrecarga de informações, nos anos 90 iniciou-se a pesquisa na área de filtragem de conteúdo. O ponto de partida foi a observação que as pessoas usam, no dia-a-dia, dicas de outros para tomar decisões. Os primeiros sistemas de

recomendação eram algoritmos capazes de analisar tendências dentro de uma comunidade e então fazer sugestões aos seus membros. Esse método é conhecido como filtragem colaborativa e foi aprimorado desde então, sendo até hoje bastante popular. Além deste, também é bastante difundido o método baseado em conteúdo, no qual novos itens são recomendados baseado no conteúdo consumido pelo usuário no passado [Ricci et al. 2011].

Neste trabalho, o foco foi sistema de recomendação guiado por estimativa (ou heurística) de confiança. Confiança é um sentimento que pode ser descrito como o ato de contar com a credibilidade e consistência das ações de um indivíduo. Trata-se de um elemento citado por muitos autores como o mais importante para a construção de uma relação de trabalho positiva, por ser essencial para criticismo construtivo e evolução mútua dentro do processo de pesquisa [Bagshaw et al. 2007]. Os dados de analise tem origem na plataforma Lattes. Para a elaboração do sistema proposto, é preciso: estudar funções e aplicações de sistemas de recomendação; modelar a rede de confiança da comunidade científica; estabelecer métricas de confiança para os dados disponíveis; estimar a confiança entre os pesquisadores; pré-selecionar recomendações; filtrar a pré-seleção com a confiança computada. Para solucionar o problema explanado, propõe-se descrever uma rede de colaborações a partir de publicações em conjunto, discutida na Seção 3.1 utilizando técnicas encontradas na literatura para computar a propagação de confiança na rede. A partir disso, é discutida a pré-seleção dos itens com o método baseado em conteúdo (Seção 3.2), proposta uma arquitetura para o sistema de recomendações (Seção 3.3).

2. Revisão Bibliográfica

Nesta seção, são apresentados os itens que compõem o embasamento teórico usado.

2.1. Sistemas de Recomendação

Os itens recomendados pelo Sistema de Recomendação (SR) podem ser os mais variados, sendo que no geral a recomendação é uma tarefa especializada na qual um tipo de item é considerado relevante para um perfil específico de usuário. Logo, a metodologia usada para sua construção, interface de usuário e critério para ordenar os resultados devem ser adaptados às especificidades da tarefa em questão [Ricci et al. 2011].

O resultado obtido por um SR é dependente da realização de uma predição, itens são apresentados ao usuário porque o sistema antecipa que sejam relevantes para ele [Ricci et al. 2011]. Geralmente na elaboração de sistemas de recomendação lida-se com usuários, denotados por $u_1, ...u_n \in U$, itens, denotados por $i_1, ...i_n \in I$ e relações, que associam usuários e itens de diversas maneiras [Ekstrand and Konstan 2019]. As associações podem ser representadas por ontologias [Primo and Loh 2006] ou no caso de relações entre usuários e itens através de uma matriz de associação $|U| \times |I|$. Assume-se a existência no mundo real de uma função $f(u, i)$ que retorne um número real representado a utilidade do item i ao usuário u. Em técnicas de filtragem colaborativa, este numero é visto como a avaliação do usuário. A tarefa do SR neste contexto é computar uma função $\hat{f}(u, i)$ que se assemelhe ao máximo à f. Assim, é possível realizar a predição de relevância de um grupo de itens para determinado usuário $\hat{f}(u_n, I)$ e recomendar os itens melhores classificados pelo SR, efetivamente filtrando o conteúdo e oferecendo ao usuário uma seleção personalizada de itens [Ricci et al. 2011].

A forma mais simples do resultado de um SR é uma lista de itens ordenada de

acordo com a preferência do usuário. A satisfação com as recomendações pode ser coletada explicitamente, como por exemplo por meio de avaliações, ou implicitamente com inferências baseadas no comportamento do usuário perante aos itens oferecidos. Para oferecer recomendações, é preciso analisar uma base de conhecimento, realizar um trabalho de classificação dos itens ofertados e então coletar algum tipo de *feedback* perante o resultado que deve ser usado para aprimorar o sistema [Shani and Gunawardana 2011].

Técnicas de filtragem colaborativa analisam o perfil do usuário e sua avaliação dos itens previamente acessados para chegar em recomendações. Procura-se achar um *cluster* de usuários com perfis similares (vizinhos). A ideia é que os itens bem avaliados pelos vizinhos serão também avaliados positivamente pelo usuário alvo, já que os perfis são semelhantes. Um problema encontrado na técnica é o da *primeira avaliação*: quando há um item novo, sem nenhuma avaliação, como saber se determinado usuário irá avaliar positivamente o mesmo? Nenhum de seus vizinhos fez avaliações [Ricci et al. 2011]. SR baseados em filtragem colaborativa são os mais populares na área e vêm sido pesquisados há mais tempo [Ricci et al. 2011]. É comum utilizar métodos baseados em vizinhança, nos quais um algoritmo de clusterização tal como o algoritmo KNN (*K-Nearest Neighbours*) é usado para determinar grupos de usuários [da Rosa Furlan et al. 2018].

O método baseado em conteúdo parte da ideia de que usuários têm interesse em itens semelhantes àqueles que lhe foram uteis no passado [Ricci et al. 2011]. No caso, é importante determinar a semelhança entre itens para então recomendar itens semelhantes aos que foram previamente bem avaliados pelo usuário. Nesse método é preciso estabelecer estratégias para descrever itens, bem como para montar o perfil dos usuários descrevendo os tipos de itens que ele tem interesse. Deve ser feito o comparativo dos itens com o perfil do usuário para predizer seu interesse. Geralmente procura-se dividir o universo dos itens, I, em categorias: relevantes ou irrelevantes. Para construir a classificação dos itens é possível usar uma série de algoritmos que realizam trabalho de classificação estatística, como por exemplo árvores de decisão. [Pazzani and Billsus 2007].

Conforme [Sinha and Swearingen 2001], estudos indicam que os usuários têm a tendência de valorizar mais as recomendações de amigos do que aquelas feitas por outros usuários com perfil semelhante, porém desconhecidos e a qualidade das recomendações de amigos superam inclusive as feitas por sistemas de recomendação. A partir deste conceito, com a grande aderência de usuários à redes sociais um novo método para a construção de sistemas de recomendação está sendo estudado, trata-se do método baseado em confiança, ou sistema de recomendação social [Ricci et al. 2011]. A construção de SR sociais depende do estabelecimento de uma rede de confiança, rede que descreve o nível de confiança entre seus membros. Assim, o usuário recebe recomendações de itens avaliados positivamente por usuários em sua rede de confiança. Estes SR usam o conceito de agregação e dissipação de confiança, ou seja, dado um grupo de usuários $u_1 \dots u_n$, calcular o nível de confiança entre u_1 e u_n considerando usuários intermediários $u_2 \dots u_{n-1}$ que possuem alguma relação de confiança, direta ou indireta, com u_1 e u_n (dissipação) ou combinar uma série de estimativas de confiança em um valor final (agregação) [Victor et al. 2011]. Um ponto fraco de tais sistemas é que a recomendação é geralmente mais previsível e pode facilmente ser inundada por itens que o usuário já conhece, enquanto técnicas mais usuais de recomendação podem apresentar resultados mais inesperados, mas relevantes ao usuário [Sinha and Swearingen 2001].

Métodos híbridos propõem a combinação de mais de um método de recomendação dentro de um sistema. É necessário para complementar técnicas que podem apresentar problemas em determinadas situações ou para oferecer resultados melhores aos usuários. Furlan [da Rosa Furlan et al. 2018] combinou os métodos baseado em conteúdo e filtragem colaborativa para solucionar o problema da primeira avaliação. Já Massa [Massa and Avesani 2004] sugere que um método que leve em consideração a confiança entre usuários pode melhorar a performance de sistemas de filtragem colaborativa.

2.2. Análise de Dados em Redes Sociais

Pode-se pensar na rede de colaborações como sendo um grafo, os nodos são pesquisadores e as arestas publicações em conjunto. Além disso uma colaboração em publicações é um voto de confiança entre os pesquisadores envolvidos. A matriz de adjacência pode ser usada para computar a propagação de confiança por meio da rede. A confiança estimada de determinado pesquisador pode levar em consideração o nível de confiança estimado dos pesquisadores que colaboraram com ele. Outro fator que pode ser considerado é a facilidade de colaboração, levando em consideração "distâncias" na rede: se dois pesquisadores A e B têm laços de confiança com um intermediário C, a colaboração entre A e B tende a ser mais fácil do que se houvesse mais intermediários na rede.

O algoritmo PageRank [Page et al. 1999] foi inspirado em parte por estudos realizados em redes de citações acadêmicas, nas quais a relevância de um artigo era descrita por contagem de citações, por exemplo. Trata-se de um método para computar um *ranking* global de citações, pensado para obter a importância das páginas web. O *ranking* R de uma página é definido como a soma dos *rankings* das páginas que oferecem *links* para ela, ponderada pelo total de *links* encontrados nas páginas. O algoritmo funciona da seguinte forma: é definido para cada página u um conjunto F_u de páginas as quais u referencia e um conjunto B_u de páginas que fazem referência à u. Sendo $\hat{A}$ a matriz de adjacência da web, tal que

$$\hat{A}_{i,j} = \begin{cases} 1 & \text{se há links de i para j} \\ 0 & \text{se não há links de i para j} \end{cases} \tag{1}$$

A matriz A deve ser obtida dividindo todas as linhas de $\hat{A}$ por $|F_u|$ (o grau do nodo u). Assim, PageRank pode ser definido como $R = c(AR + E)$, sendo c um fator de normalização.

Quando ocorrem ciclos no fluxo de referência, nos quais duas páginas se referenciam mutuamente e não fazem referência a nenhuma outra página, pode ocorrer o chamado *rank sink*: referências exteriores injetam *ranking* no ciclo, fazendo com que páginas do ciclo acumulem pontuação, porém sem distribuição. Para solucionar, foi introduzido o vetor E, que no modelo de PageRank é o conceito de um *random surfer*, ou seja, uma heurística representando a probabilidade de um usuário da internet aleatoriamente mudar a página, sem seguir nenhum de seus *links* [Page et al. 1999].

Já a centralidade è uma métrica da teoria dos grafos usada para representar a importância de um nodo na rede. Centralidade de grau è definida como o numero de arestas com as quais um nodo se conecta. A métrica de centralidade apresentada em [Opsahl et al. 2010] se encaixa particularmente bem em casos nos quais o peso da aresta

representa a força da conexão, tal qual o problema proposto neste trabalho, por incorporar simultaneamente o grau (número de conexões) e a força (os pesos de cada conexão) dos nodos. O peso pode ser a soma das relevâncias das obras publicadas em conjunto entre os pesquisadores. A fórmula proposta pelos autores faz isso, definindo um parâmetro α para ajustar a importância de grau e força:

$$C_D^{w\alpha}(i) = k_i \times \left(\frac{s_i}{k_i}\right)^{\alpha} = k_i^{1-\alpha} \times s_i^{\alpha} \tag{2}$$

onde k e o s são os vetores com os valores do peso das arestas, sendo o k o vetor ponderado, e o vetor s contendo valores 1 ou 0.

No contexto de distância, o algoritmo de Dijikstra [Dijkstra 1959] é definido para calcular distâncias em redes nas quais os pesos representam o custo de travessia. O trabalho de Opsahl e Skvoretz [Opsahl et al. 2010] é definido em redes onde os pesos (w) representam força dos laços, então os autores sugerem que os pesos devem ser invertidos. Além disso, o objetivo do trabalho é considerar também o número de nós intermediários, então os autores propõem novamente o uso de um parâmetro de ajuste, α, que controla o quão importante considera-se o número de nodos intermediários e a força das conexões.

$$d^{w\alpha}(i,j) = min\left(\frac{1}{(w_{ih}^{\alpha})} + \cdots + \frac{1}{(w_{hj}^{\alpha})}\right) \tag{3}$$

Frequentemente na análise de dados é necessário o uso de clusterização. O algoritmo MeanShift é indicado para dados nos quais se espera muitos *clusters* distintos e de tamanhos variáveis [Pedregosa et al. 2011]. A implementação é descrita como uma busca por centróides baseada em grafo de vizinhos mais próximos. O parâmetro *bandwidth* é usado como estimativa para o tamanho dos *clusters*.

2.3. Trabalhos Correlatos

Os trabalhos correlatos foram escolhidos utilizando como critério a contemporaneidade e semelhança com o presente trabalho.

No trabalho de [da Rosa Furlan et al. 2018], é abordado o problema da sobrecarga de informações dos pesquisadores baseando-se no perfil do currículo Lattes. O trabalho busca recomendações de produções científicas utilizando o motor de buscas Google Acadêmico e traz uma combinação das técnicas de filtragem colaborativa e baseado em conhecimento. A metodologia para gerar recomendações utilizada neste trabalho foi usada no presente trabalho como referência para a elaboração do SR, levando em consideração os pontos fracos e fortes da abordagem descrita no trabalho. Em particular, será considerada a maneira com que o trabalho propôs solucionar o problema da avaliação inicial de um SR de filtragem colaborativa através do método baseado em conteúdo.

Em [Primo and Loh 2006], são apresentadas algumas das mais populares técnicas de recomendação, bem como a justificativa e contexto para a correta implementação. O trabalho descreve abordagens para a elaboração de um SR de obras literárias em bibliotecas digitais, usando as técnicas de filtragem colaborativa e baseado em conteúdo bem

como uma abordagem híbrida. O contexto do sistema de recomendação descrito no trabalho se assemelha ao do presente trabalho por ter como alvo uma biblioteca digital. O comparativo das metodologias usadas serve como referência para a elaboração do SR descrito no presente trabalho.

No artigo [Massa and Avesani 2004], sugere-se a possibilidade de melhorar as sugestões em sistemas de recomendação com métricas de confiança, é descrita a modelagem de uma rede de confiança e a necessidade de métricas de propagação de confiança, considerada computável em mais usuários do que a similaridade de perfis. A métrica usada é a distância mínima entre nós para a estimativa de confiança local. Os autores sugerem ainda a aplicação do algoritmo PageRank [Page et al. 1999] como métrica de confiança global. Buscou-se seguir a arquitetura sugerida no trabalho para a construção de um SR que combina os métodos baseados em conteúdo e confiança, que é composta por módulos substituíveis que representam conceitualmente a aplicação de um algoritmo. A adaptação da arquitetura está descrita na Seção 3.3.

3. Metodologia

Para chegar nas recomendações, é proposto um trabalho em dois momentos: estimar a confiança entre pesquisadores e selecionar potenciais colaboradores baseando-se no perfil dos pesquisadores. A seleção é filtrada e ordenada de acordo com o nível de confiança estimado dos pesquisadores. O primeiro passo é modelar uma rede de confiança da comunidade científica, descrita por autores e publicações. Pode então ser feita uma seleção dos pesquisadores cadastrados com base no perfil do usuário alvo e ordenar a seleção de acordo com o nível de confiança de cada pesquisador. A confiança pode ser local ou global, sendo que local diz respeito à confiança estimada de um pesquisador específico em seus colegas e a global corresponde à confiança da comunidade em cada pesquisador.

3.1. Computando confiança

A Figura 1 representa a rede de confiança proposta (discutida na Seção 2.2, ilustrando os pesos das arestas (discutidos na Seção 3.1.1).

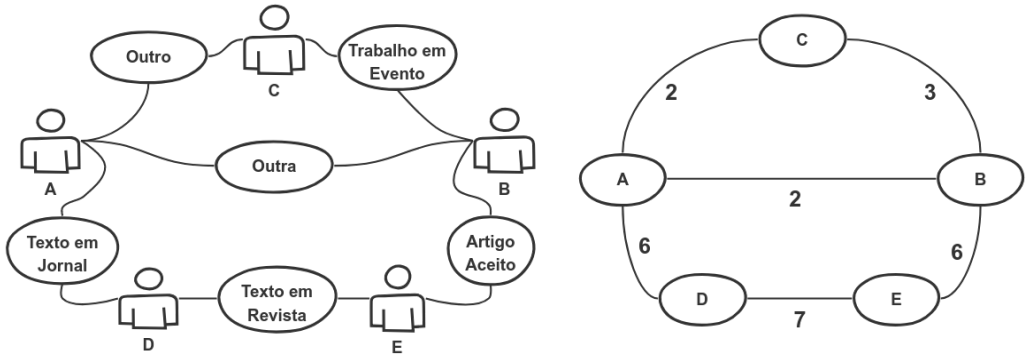

Figura 1. Rede de confiança

A aplicação do PageRank (discutido na Seção 2.2) em um grafo não-direcionado gera um vetor R estatisticamente similar à distribuição de grau dos nodos da rede [Perra and Fortunato 2008]. Isto é, aplicando diretamente o algoritmo ao problema proposto, no final das contas a confiança seria proporcional ao número de publicações do

autor (**centralidade** do nodo). Enquanto esta métrica é relevante, perde-se a ideia inicial: não é considerada a confiança dos colaboradores, somente o valor total de colaborações. Além disso, dois conceitos importantes não são levados em consideração: a relevância e o número de colaborações entre os pesquisadores. No caso da **relevância**, a importância da publicação é uma dica para o nível de confiança mútua entre os pesquisadores: colaborações em publicações importantes requerem maior confiança. O total de **colaborações em conjunto** entre um par de pesquisadores, por sua vez, indica uma relação mais duradoura, com mais confiança mútua. É importante distinguir total bruto de publicações de determinado pesquisador e o número de colaborações entre dois pesquisadores, pois há mais confiança quando observa-se frequentes colaborações. Uma vez estabelecida uma heurística para a importância de determinada colaboração, pode-se usar a importância como heurística para avaliar a força dos laços de confiança.

3.1.1. Relevância, Centralidade e Distância

As publicações não são iguais entre si. Para construir uma heurística ou estimativa que defina a relevância de uma publicação deve-se considerar as características descritas na representação da mesma, atribuindo pesos aos seus atributos. Aqui, a heurística considerada é a natureza da publicação, com pesos atribuídos conforme a Tabela 1.

Tabela 1. Heurística de Relevância.

Tipo de Publicação	Peso
Livro Publicado ou Organizado	9
Capítulo de Livro Publicado	8
Artigo Publicado	8
Artigo Aceito para Publicação	7
Texto em Jornal/Revista	6
Trabalho em Evento	3
Outra Produção Bibliográfica	2
Prefácio/Posfácio	2
Tradução	1

Vale ressaltar que a heurística neste caso é relativa por haver muitos fatores que influenciam a relevância de uma publicação: artigos em certas publicações de prestígio podem valer mais que capítulos de livro, e o mesmo pode ser verdade para textos em jornais ou revistas. Da mesma maneira, publicações mais recentes podem valer mais do que publicações mais antigas, porém o contrário pode ser verdade para linhas de pesquisa na área da história, por exemplo. É possível também considerar mais do que a relevância e quantidade das publicações para descrever os laços de confiança entre pesquisadores.

Considerando as heurísticas discutidas, é possível alcançar um fluxo de confiança mais interessante na rede modelada. O conceito de agregação e dissipação de confiança pode seguir a ideia do algoritmo PageRank aplicando-se um algoritmo para o cálculo da centralidade dos nodos (Equação 2). No caso, a relevância das publicações e a quantidade de colaborações devem ser levadas em consideração na distribuição de confiança. Ao incorporar o número e a relevância das publicações como um peso para as arestas da rede,

é reintroduzido o conceito de considerar a confiança da comunidade nos colaboradores que depositaram confiança em determinado autor através de publicações em conjunto para o cálculo da confiança estimada de tal autor. Assim, é possível chegar em um fluxo de confiança apurado levando em conta informações sobre obras e autores que são relevantes para considerar a confiança compartilhada entre os membros da rede.

Na Tabela 2 são apresentados os diferentes valores computados para a centralidade de cada nodo aplicando-se a Equação 2 na rede exemplificada na Figura 1, variando o parâmetro α. No caso, as recomendações seriam ordenadas do maior para o menor valor computado. Para $\alpha = 0$ a centralidade de cada pesquisador seria igual ao número de colaborações, aumentando-se α é atribuída maior importância para a heurística de relevância das publicações.

Tabela 2. Centralidade

Nodo	$C_D^{w\alpha}(i)$			
	$\alpha := 0.00$	$\alpha := 0.50$	$\alpha := 1.00$	$\alpha := 1.50$
A	3.00	5.47	10.00	18.25
B	3.00	5.74	11.00	21.06
C	2.00	3.16	5.00	7.90
D	2.00	5.09	13.00	33.00
E	2.00	5.09	13.00	33.00

A centralidade do pesquisador, ponderada pelo número de colaborações e suas relevâncias se mostra em teoria uma forte métrica de confiança global. O cálculo de confiança local, porém, oferece uma estimativa da **confiança subjetiva** de um usuário em relação aos membros da rede. Outra métrica valiosa neste contexto é a distância entre os pesquisadores, isto é, identificar **amigos de amigos** e pesquisadores próximos na rede é uma maneira de promover a colaboração entre pesquisadores: pesquisadores próximos na rede podem encontrar mais facilidade para realizar colaborações. Para tal, a relevância e quantidade de colaborações (pesos das arestas - confiança) deve ser um fator positivo e o número de nodos intermediários entre os autores um fator negativo (maior distância).

O conceito de distância (discutido na Seção 2.2 e ilustrado na Equação 3), aplicado à rede de colaborações, significa que a distância entre os pesquisadores levará em consideração o nível de confiança das conexões bem como o número de pesquisadores intermediários. Para $\alpha < 1$, caminhos com maior número de intermediários são considerados mais distantes, enquanto $\alpha > 1$ vai considerar mais importante a força das relações de confiança, atribuindo menores distâncias para caminhos onde há fortes relações de confiança entre os pesquisadores, podendo estes ter mais intermediários [Opsahl et al. 2010]. Na Tabela 3 são apresentadas as distâncias calculadas entre os pesquisadores A e B, conforme a rede exemplificada na Figura 1. As recomendações seriam ordenadas da menor para a maior distância.

3.2. Pré-seleção de Recomendações

Com as métricas de confiança propostas, é possível estabelecer níveis de confiança da comunidade, uma rede subjetiva do pesquisador alvo ou distâncias ponderadas por confiança e usar as predições para oferecer sugestões de colaboradores para cada membro da rede em diferentes contextos. Todavia, confiança apenas pode não ser suficiente

Tabela 3. Distâncias

Caminho	$d^{w\alpha}(i,j)$			
	$\alpha := 0.00$	$\alpha := 0.50$	$\alpha := 1.00$	$\alpha := 1.50$
{A, B}	1.00	0.81	0.50	0.35
{A, C, B}	2.00	1.53	0.83	0.54
{A, D, E, B}	3.00	1.72	0.47	0.19

para recomendações de qualidade. Considerar também a linha de pesquisa do pesquisador no momento e os perfis dos potenciais colaboradores pode aumentar a qualidade das recomendações: mesmo que a confiança em um pesquisador seja alta, a sugestão de colaboração pode não fazer sentido caso as linhas de pesquisa não se encaixem.

Para aprimorar as recomendações, é proposta uma pré-seleção dos itens utilizando um método baseado em conteúdo. A técnica consiste no cálculo de correspondência de palavras chave em um modelo de espaço vetorial (MEV), visto que esta é a mais comum em sistemas de recomendação baseados em conteúdo [Ricci et al. 2011]. O perfil do pesquisador é representado por um vetor em um espaço *n-dimensional*: $d_j = w_{1,j}, w_{2,j}, \ldots, d_{n,j}$, no qual $w_{i,j}$ representa o quanto o termo i é relevante dentro do trabalho do pesquisador j. Pode-se pensar em uma matriz na qual as linhas são pesquisadores, conforme descrito e as colunas representam os termos-chave extraídos do universo das publicações (*corpus*), removendo palavras vazias - "ou", "de", "para" ... - tanto em português quanto em inglês.Tal matriz é construída por meio da técnica de vetorização *TF-IDF*, na qual considera-se termos importantes aqueles que aparecem com frequência relacionados á um item específico e com menor frequência nos outros itens do *corpus* [Pazzani and Billsus 2007].

A partir disso, é preciso computar a semelhança entre termos. Para tal, a proposta é o uso da similaridade de cossenos por ser a técnica mais comumente aplicada [Ricci et al. 2011]. Para a pré-seleção dos itens, a proposta é basear-se em uma *query* que representa o trabalho sendo desenvolvido pelo pesquisador no presente.

3.3. Arquitetura

Nesta seção, detalha-se de como é possível combinar as métricas de confiança e o método baseado em conteúdo para obter um resultado.

Em [Massa and Avesani 2004], é sugerida uma arquitetura de SR combinando filtragem colaborativa e método baseado em confiança. O sistema é descrito em módulos substituíveis e, portanto, pode ser usado para combinar os métodos baseado em conteúdo e em confiança. Basicamente, conforme apresentado na Figura 2, a saída do método usado para pré-seleção é usada para filtrar e ordenar as recomendações a partir das confianças computadas.

É preciso então definir um método para oferecer ao usuário uma amostra seleta com os pesquisadores mais relevantes seguindo as métricas de confiança (filtragem por confiança) a partir de uma lista de itens ordenada por similaridade de cossenos e vetores de confiança estimada. As linhas em pontilhado da Figura 2 representam que os objetos são intercaláveis: pode-se usar um ou outro ou combinações. Após a pré-seleção, o custo computacional da aplicação das distâncias, por exemplo, diminui consideravelmente pois

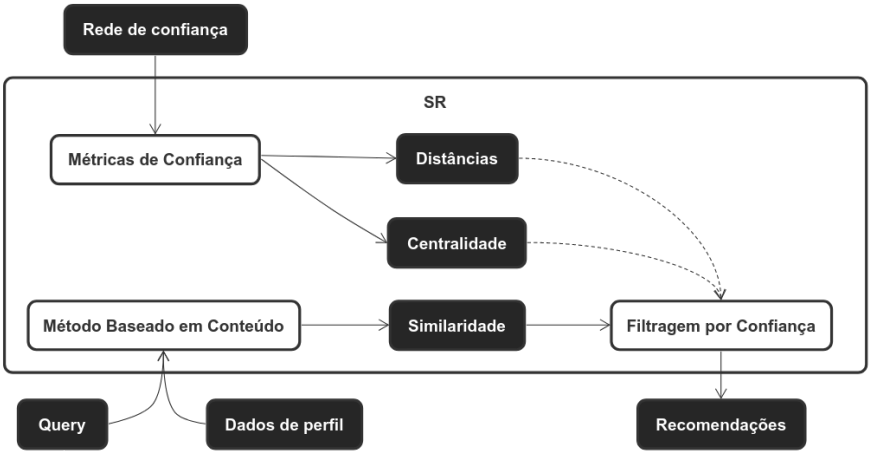

Figura 2. Arquitetura do Sistema de Recomendação proposto.

é preciso apenas computar distâncias entre o pesquisador alvo e uma amostra pequena de pesquisadores com o perfil compatível com a *query*. Portanto, uma vez aplicado o método por conteúdo, pode-se aplicar as métricas de confiança em uma amostra reduzida da rede.

3.4. Tecnologias

Os dados usados no trabalho são provenientes do banco de dados relacional da Plataforma Kennis (www.kennis.com.br), que extrai os dados dos currículos de pesquisadores cadastrados na Plaforma Lattes com o uso de um *parser*, cuja versão inicial é descrita em [Prass et al. 2019]. Optou-se pelo uso dos dados da Plataforma Kennis e não pelos dados originais da Plataforma Lattes, pois durante o processo de *parsing* dos currículos, a Plataforma Kennis faz a limpeza e o pré-processamento dos dados do website do curriculo Lattes, associando pesquisadores e suas publicações, conforme mostra a Figura 3.

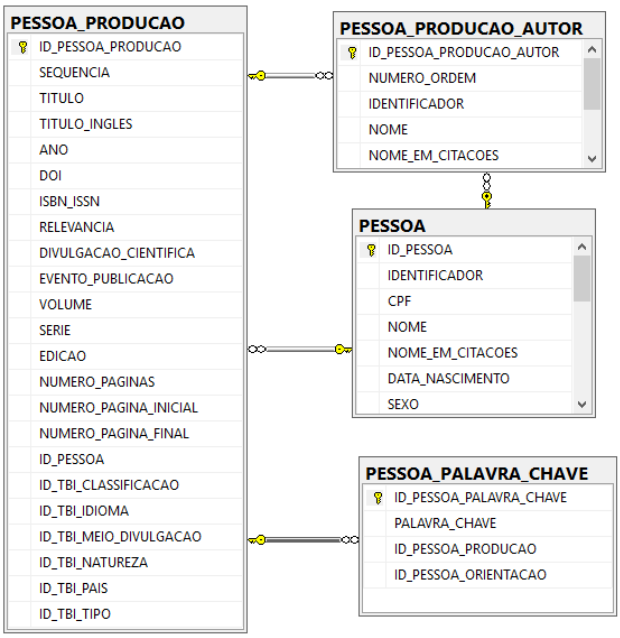

Figura 3. Tabelas Pessoa, Produção e associativa [Prass et al. 2019].

A partir do banco, foi possível construir uma base de conhecimento focada especificamente no objetivo do trabalho. Foi gerado a partir dos dados da Kennis um arquivo com dados como o titulo da obra, a ID de seu autor, seu tipo (os possíveis tipos de obras são ilustrados na Tabela 1), palavras chaves e o contexto de sua publicação. Os dados exportados da plataforma Kennis resultaram em um arquivo *CSV - comma separated value* com 16793 linhas, cada uma representando uma publicação cadastrada por seu autor.

Foi usada a linguagem Python que possui recursos para trabalhos relacionados a manipulação de dados e computação numérica. A biblioteca pandas facilita a descoberta de conhecimento em bancos de dados, oferecendo objetos que encapsulam uma base de dados e ajudam na sua manipulação. Aliado com a biblioteca *scikit-learn* que implementa diversos algoritmos usados na mineração de dados e computação científica (como vetorização TF-IDF e o algoritmo de clusterização MeanShift [Pedregosa et al. 2011]), também foi utilizada a biblioteca numpy para computação numérica. O pacote *NetworkX* oferece um ambiente de programação (API) para a criação, manipulação e estudo de redes complexas, com implementação de algoritmos de análise de grafo capazes de tratar grandes *datasets*, devido ao uso de linguagens como o C++ na implementação dos algoritmos [Hagberg et al. 2008]. A distribuição *Anaconda* funciona como um gerenciador de pacotes e ambientes da linguagem Python e por padrão oferece todas as bibliotecas citadas, além da interface *jupyter*, que facilita a exploração de dados por via de uma interface gráfica acessível via navegador.

Para execução do código, foram usados diversos serviços de computação em nuvem variando nas seguintes características: hardware, preço, disponibilidade de tempo e facilidade de uso. A plataforma *datalore* não possui limite de uso e é capaz de proporcionar um ambiente anaconda para usuário cadastrados depois de poucos cliques (sem custos). Porém, possuía apenas instâncias com 4GB de memória RAM e uma CPU de 2 *cores* e não está disponível a todo o tempo, apresentando quedas do servidor que embora raras impossibilitam a total dependência da plataforma. Devido a essas limitações, a plataforma foi usada para análise exploratória dos dados e desenvolvimento dos algoritmos usando uma amostra reduzida dos dados. Para computações custosas usando a base completa, foi usada a plataforma *salamander.ai*, que oferece facilidade semelhante a da plataforma *datalore* para o *deploy* de instâncias prontas para o uso. A plataforma não possui modalidades de graça e é preciso adicionar créditos, que são deduzidos da conta do usuário conforme o tempo de uso e ao hardware selecionado. Foi usada uma instância de 36 núcleos otimizados para computação e 72 gigabytes de memória RAM, ao custo de 93 centavos de dólar por hora. Por fim, o *Google cloud platform* (GCP) oferece opções de hardware para seleção do usuário de até 96 núcleos, com um crédito inicial de 300 dólares de bônus. Porém não oferece nenhuma configuração prévia, exigindo o usuário fazer a configuração do ambiente. No GCP foi usado 16 núcleos de CP otimizados para computação e 32 gigabytes de RAM, pois observou-se que o uso de CPU era limitado pela quantidade de RAM disponível.

4. Resultados e Discussões

Nesta seção, buscou-se apresentar alguns resultados relevantes e já discuti-los, para facilitar o entendimento da proposta e dos próprios resultados.

4.1. Pré-Processamento

Os dados originalmente contém 169 autores e 15821 títulos de obra distintos. É importante notar que os títulos distintos não representam publicações distintas, pois é possível que dois pesquisadores cadastrem a mesma publicação com títulos levemente diferentes (devido a erros de caligrafia, supressão de artigos, distinções de acentuação e de letras maiúsculas ou minúsculas), o que pode causar com que a mesma publicação apareça diversas vezes com pequenas variações de título. Por isso, na fase de pré-processamento se fez necessário agrupar as publicações em comum cadastradas por pesquisadores distintos.

Foi gerada uma matriz TF-IDF usando a classe *TfidfVectorizer* de *scikit-learn*, que resultou em uma tabela com todos os termos que apareciam nos títulos dos artigos, entretanto, foi necessário excluir as palavras vazias (artigos, preposições e outros). Essa matriz tinha as 17 mil linhas da base original, mas 20 mil colunas, referentes aos termos dos títulos. O valor da linha coluna era a importância do termo para o artigo.

Para agrupar as publicações foi usado o algoritmo *MeanShift*, conforme exposto pela Seção 2.2, na matriz TF-IDF dos títulos das obras. Foram feitos testes com o parâmetro *bandwidth* variando de 0.3 até 1.0, procurando um valor no qual os *clusters* resultantes correspondem a publicações em comum cadastradas por pesquisadores distintos. Notavelmente, valores mais altos acabaram por agrupar títulos diferentes em um único *cluster*, efetivamente resultando em falsos positivos. Observou-se que o valor 0.3 resultou em *clusters* nos quais os títulos eram efetivamente os mesmos, diferindo principalmente em detalhes como acentuação ou supressão de artigos. Depois da clusterização, foram filtrados os dados para remover as publicações que não foram agrupadas em nenhum *cluster* e *clusters* nos quais todas as publicações são pertencentes a um mesmo autor, pois representam arestas da rede que não conectam dois autores.

```
bandwidth 03: (3502, 14)
bandwidth 04: (3702, 14)
bandwidth 05: (4026, 14)
bandwidth 06: (4524, 14)
bandwidth 07: (5204, 14)
bandwidth 08: (6106, 14)
bandwidth 09: (7279, 14)
bandwidth 10: (14139, 14)
```

Figura 4. Numero de linhas, colunas do arquivo *CSV* depois da filtragem

A Figura 5 exemplifica um *cluster* obtido usando um *bandwidth* de 0.3. Pode-se notar que embora tenha pequenas variações no título, a publicação de ID (*pubid* na tabela) 26 trata-se de um trabalho publicado em evento no ano 2015 cadastrado pelos autores $6, 9, 70$ e 128 (coluna $authid$). A Figura 4 mostra a quantidade de linhas dos dados depois da filtragem. Por exemplo, o cluster *bandwidth* 03 possui 3502 publicações e 14 colunas após a filtragem. A biblioteca Pandas foi usada para a filtragem dos dados. Foi definida uma função not_single, que retorna as publicações que não são únicas, e tambem uma função $shared\_pubs$, que aplica not_single aos dados e permite somente os *clusters* com mais de uma publicação.

authid	pubid	TIPO_PRODUCAO	ANO	TITULO
111	25	Artigo Publicado	2010.0	Curso de Enfermagem da UNIFRA: avanços e conquistas em uma histór...
6	26	Trabalho em Evento	2015.0	UMA ABORDAGEM ORIENTADA A OBJETOS APLICADA À AUTOMAÇÃO',
9	26	Trabalho em Evento	2015.0	UMA ABORDAGEM ORIENTADA A OBJETOS APLICADA À AUTOMAÇÃO
70	26	Trabalho em Evento	2015.0	Uma abordagem orientada a objetos aplicada à automação
128	26	Trabalho em Evento	2015.0	Uma Abordagem Orientada a Objetos Aplicada à Automação

Figura 5. Exemplo de um cluster com bandwidth 0.3.

4.2. Estimativa de confiança

Foi usado o tipo *DiGraph* presente em *NetworkX* para a modelagem da rede de confiança. O tipo representa um grafo não-direcionado com a possibilidade de múltiplas arestas entre dois nodos. Assim, para cada publicação em comum entre dois pesquisadores, foi adicionada uma aresta entre eles com o peso equivalente aos das heurísticas apresentadas na Tabela 1. Para a atribuição da heurística na formação do grafo, os dados pré-processados foram lidos de um arquivo *CSV* resultante da fase de pre-processamento, e então foi adicionada uma coluna numérica com o valor correspondente ao tipo de cada publicação.

Foram então definidas funções para a geração da rede de confiança. Destaca-se a função *edge* que retorna as publicações em comum e foi usada para popular o grafo da rede de confiança. A Figura 6 mostra as funções que calculam as equações de distância e centralidade apresentadas na seção de revisão bibliográfica, a função $path\_length$ implementa a Equação 3 e a função $degree\_centrality$ implementa a Equação 2.

```python
import numpy as np

def path_length(path_weights, alpha):
    return sum([1 / weight ** alpha for weight in path_weights])

def jumps(l, n=2):
    return [l[i:i+n] for i in range(len(l))][:-1]

def sum_weights(n, j):
    return sum([n[j[0]][j[1]][k]['weight'] for k in n[j[0]][j[1]]])

def sum_of_weights(jumps, network):
    return [sum_weights(network, jump) for jump in jumps]

def deg_arr(deg):
    return np.array([v for _, v in deg])

def degree_centrality(network, alpha):
    k = deg_arr(network.degree()) ** (1 - alpha)
    s = deg_arr(network.degree(weight='weight')) ** alpha
    centralities = k * s
    return [{node: centralities[i]} for i, node in enumerate(network.nodes)]
```

Figura 6. Estimativa de confiança via centralidade e distancias.

A Figura 7 (a) apresenta o resultado das estimativas de confiança por distância entre os pesquisadores $u_1 := 14 \ldots u_n := 40$. A esquerda observa-se uma lista representando o caminho percorrido e a direita o valor da distancia. Pode-se observar que embora a biblioteca *NetworkX* gere caminhos repetidos, eles possuem o mesmo valor para a distância pois o algoritmo leva em consideração o numero de arestas entre os nodos [Opsahl et al. 2010]. Por apresentar um número muito grande de caminhos com mui-

```
[([14, 25, 125, 39, 118, 89, 40], 1.8838118415212128),
 ([14, 25, 125, 39, 118, 89, 40], 1.8838118415212128),
 ([14, 25, 125, 39, 118, 89, 40], 1.8838118415212128),
 ([14, 25, 125, 39, 118, 89, 40], 1.8838118415212128),
 ([14, 25, 125, 39, 118, 89, 40], 1.8838118415212128),
 ([14, 25, 125, 39, 118, 89, 40], 1.8838118415212128),
 ([14, 25, 125, 39, 144, 89, 40], 1.4528987074039548),
 ([14, 25, 125, 39, 144, 89, 40], 1.4528987074039548),
 ([14, 25, 125, 39, 144, 89, 40], 1.4528987074039548),
```

(a) Distancias

```
[{1: 236.92825918408298},
 {31: 226.8633950199988},
 {108: 210.35683967962632},
 {162: 184.0217378463751},
 {0: 183.25665062965658},
 {36: 180.32747988035544},
 {53: 179.55500549970753},
 {106: 143.10835055998655},
 {111: 133.0413469565007},
 {129: 129.18204209564112},
 {86: 127.67928571228772},
```

(b) Centralidade

Figura 7. Estimativas de confiança com $\alpha := 0.5$

tos intermediários no grafo, foi necessário realizar uma poda (parâmetro *cutoff*). Na Figura 7 (b) observa-se a computação das as centralidades de cada pesquisador, ou seja, a estimativa de confiança global. Cada item da lista representa um par $\{chave : valor\}$, sendo a chave a ID do pesquisador e o valor a sua centralidade. Destaca-se que o tempo de processamento desse processo foi menor que o processo de cálculo de distâncias, e também que a métrica centralidade pode ser re-aproveitada para múltiplas recomendações, enquanto as distancias devem ser calculadas para cada par de pesquisadores.

As métricas apresentadas podem então ser aplicadas ao resultado de uma pre-seleção qualquer, tal como a apresentada na Figura 2, para aumentar o escopo das recomendações obtidas com a dimensão confiança. Por exemplo, sendo o resultado da recomendação baseada em conteúdo uma lista ordenada de pesquisadores, pode-se re-ordenar a lista utilizando uma das métricas sugeridas, ou uma combinação delas.

```
In [30]: recommendations
Out[30]: [1, 0, 43, 50, 64, 80, 101, 66, 38]
```

```
In [25]: centralities
Out[25]: {1: 236.92825918408298,
          50: 118.94536560959406,
          0: 183.25665062965658,
          66: 50.91168824543142,
          38: 48.28043081829324,
          43: 51.16639522186412,
          101: 54.772255750516614,
          64: 9.38083151964686,
          80: 4.69041575982343}
```

```
In [27]: distances
Out[27]: {43: 0.4214985851425088,
          64: 1.2908044844199997,
          80: 0.989293139842236,
          38: 0.19245008972987526}
```

```
In [26]: centrality_recommendations
Out[26]: [1, 0, 50, 101, 43, 66, 38, 64, 80]
```

```
In [28]: distance_recommendations
Out[28]: [38, 43, 80, 64, 1, 0, 50, 101, 66]
```

Figura 8. Exemplo de Recomendações.

Para ilustrar a arquitetura discutida na Seção 3.3, definiu-se uma função que recebe uma ID de pesquisador e retorna um vetor com as IDs dos pesquisadores com perfis mais semelhantes por similaridade de cossenos conforme discutido na Seção 3.2. Na célula 30 da Figura 8, o vetor $recommendations$ representa a pré-seleção baseada em conteúdo para o pesquisador de $ID := 1$ (como o próprio pesquisador não é excluído para a seleção dos mais semelhantes, nota-se que ele é o primeiro perfil sugerido, validando a recomendação baseada em conteúdo). As células 25 e 26 respectivamente representam os valores da centralidade da pré-seleção ($centralities$) e a lista re-ordenada

de recomendações ($centrality\_recommendations$), baseando-se em centralidade. As células 27 e 28 mostram as distâncias entre o pesquisador 1 e os pesquisadores da pré-seleção para os quais existem caminhos na rede ($distances$), bem como a lista ordenada por distâncias ($distance\_recommendations$).

5. Conclusões

Neste trabalho buscou-se apresentar sistemas de recomendação para promover a colaboração entre membros de uma comunidade de pesquisadores. Foram discutidos conceito de Sistema de Recomendação, as técnicas mais utilizadas e uma tendência mais recente de recomendações baseadas em confiança. Sugeriu-se um modelo de rede de confiança na qual pesquisadores são conectados por publicações em conjunto, considerado um voto de confiança. Também foram discutidas três métricas de propagação e agregação de confiança na rede, usando conceitos do algoritmo PageRank [Page et al. 1999] e de métricas de centralidade e distância entre nodos em grafos não-direcionados ponderados [Opsahl et al. 2010].

A partir disso, foi proposta a recomendação por meio da pré-seleção de perfis baseada em conteúdo, seguida de filtragem dos perfis via estimativas de confiança, obedecendo a arquitetura proposta por Massa e Avesani [Massa and Avesani 2004] para um SR híbrido utilizando o método baseado em conteúdo.

Com relação aos resultados obtidos, destaca-se que a recomendação baseada em centralidades mostrou-se mais eficaz computacionalmente, pois os valores são calculados com rapidez e podem ser reutilizados. Enquanto as recomendações baseadas em distância devem ser calculadas par-a-par e são computacionalmente complexas, pois dependem de uma busca em grafo por (todos os) caminhos entre dois nodos. Porém, são capazes de achar amigos de amigos, o que pode influenciar a facilidade da colaboração. Nos resultados da Figura 8, observa-se que para o pesquisador 1 (que possui a maior centralidade entre todos os pesquisadores dos dados) foi possível achar distâncias para apenas 4 das 9 recomendações da pré-seleção. Sugere-se então a combinação das duas métricas aplicando a centralidade aos perfis para os quais não é possível aplicar a distância.

É possível estender o trabalho proposto pensando em melhores heurísticas para a relevância das publicações. Também se sugere como trabalho futuro o processo de validação do Sistema de Recomendação. Em [Shani and Gunawardana 2011], os autores propõem que é importante considerar o contexto da aplicação de cada SR, pois como os objetivos de diferentes sistemas variam, é natural que variem também as métricas de validação. Como o principal fator de sucesso do SR proposto é a promoção da colaboração entre os pesquisadores da rede, a métrica mais completa para a validação do sistema é subjetiva ao usuário.

Referências

Bagshaw, D., Lepp, M., and Zorn, C. R. (2007). International research collaboration: Building teams and managing conflicts. *Conflict Resolution Quarterly*, 24(4):433–446.

da Rosa Furlan, L. A., de Oliveira Zamberlan, A., Vieira, S. A. G., and Canal, A. P. (2018). Desenvolvimento de um sistema de recomendação para bibliotecas digitais. *Disciplinarum Scientia— Naturais e Tecnológicas*, 19(1):87–104.

Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1):269–271.

Ekstrand, M. D. and Konstan, J. A. (2019). Recommender systems notation: Proposed common notation for teaching and research. *arXiv preprint arXiv:1902.01348*.

Hagberg, A., Swart, P., and S Chult, D. (2008). Exploring network structure, dynamics, and function using networkx. Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States).

Massa, P. and Avesani, P. (2004). Trust-aware collaborative filtering for recommender systems. In *OTM Confederated International Conferences: On the Move to Meaningful Internet Systems*, pages 492–508. Springer.

Opsahl, T., Agneessens, F., and Skvoretz, J. (2010). Node centrality in weighted networks: Generalizing degree and shortest paths. *Social networks*, 32(3):245–251.

Page, L., Brin, S., Motwani, R., and Winograd, T. (1999). The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab.

Pazzani, M. J. and Billsus, D. (2007). Content-based recommendation systems. In *The adaptive web*, pages 325–341. Springer.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Perra, N. and Fortunato, S. (2008). Spectral centrality measures in complex networks. *Physical Review E*, 78(3):036107.

Prass, F. S., Matheus Boijink, F., and de Oliveira Zamberlan, A. (2019). Parser e leitura aautomatizada de currículos da plataforma lattes para extração de indicadores acadêmicos e tecnológicos. In *Comunicação, Mídias e Educação*, pages 492–508. Atena Editora.

Primo, T. and Loh, S. (2006). Técnicas de recomendação para usuários de bibliotecas digitais. *Simpósio Brasileiro de Sistemas de Informação. Curitiba, PR*.

Ricci, F., Rokach, L., and Shapira, B. (2011). Introduction to recommender systems handbook. In *Recommender systems handbook*, pages 1–35. Springer.

Shani, G. and Gunawardana, A. (2011). Evaluating recommendation systems. In *Recommender systems handbook*, pages 257–297. Springer.

Sinha, R. R. and Swearingen, K. (2001). Comparing recommendations made by online systems and friends. In *DELOS*.

Victor, P., De Cock, M., and Cornelis, C. (2011). Trust and recommendations. In *Recommender systems handbook*, pages 645–675. Springer.

Ware, M. and Mabe, M. (2015). The stm report. *International Association of Scientific, Technical and Medical Publishers*, page 5.