USO DE INFORMAÇÕES ESPECÍFICAS DE DOMÍNIO EM RECOMENDAÇÕES PARA TURISMO

RUHAN BIDART

USO DE INFORMAÇÕES ESPECÍFICAS DE DOMÍNIO EM RECOMENDAÇÕES PARA TURISMO

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação.

Orientador: Adriano César Machado Pereira
Coorientadora: Jussara Marques Almeida

Belo Horizonte
Agosto de 2015

Bidart, Ruhan

B584u Uso de Informações Específicas de Domínio em Recomendações para Turismo / Ruhan Bidart. — Belo Horizonte, 2015
xvi, 63 f. : il. ; 29cm

Dissertação (mestrado) — Universidade Federal de Minas Gerais
Orientador: Adriano César Machado Pereira
Coorientadora: Jussara Marques Almeida

1. Computação-Teses. 2. Sistemas de Recomendação. 3. e-turismo. I. Título.

CDU 519.6*73(043)

UNIVERSIDADE FEDERAL DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

FOLHA DE APROVAÇÃO

Uso de informações específicas de domínio em recomendações para turismo

RUHAN BIDART

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:

PROF. ADRIANO CÉSAR MACHADO PEREIRA - Orientador
Departamento de Ciência da Computação - UFMG

PROFA. JUSSARA MARQUES DE ALMEIDA GONÇALVES - Coorientadora
Departamento de Ciência da Computação - UFMG

PROF. ANISIO MENDES LACERDA
Departamento de Computação - CEFET

PROF. MARCELO GARCIA MANZATO
Departamento de Ciências da Computação - USP

PROF. MARCOS ANDRÉ GONÇALVES
Departamento de Ciência da Computação - UFMG

Belo Horizonte, 21 de agosto de 2015.

Resumo

Sistemas de Recomendação desempenham um papel chave no processo de decisão dos usuários em sistemas Web. No turismo, esses sistemas são largamente utilizados na recomendação de hotéis, atrações turísticas, acomodações, etc. Mais recentemente, sistemas de recomendação têm sido utilizados para prover serviços que auxiliem os usuários a criar seu plano de viagens. Um plano de viagens consiste de vários estágios, tais como: (i) escolha dos destinos, (ii) seleção de atrações turísticas, (iii) escolha de acomodações, (iv) escolha de rotas, entre outros. Esta dissertação possui como foco primeiro estágio, que consiste em recomendar um conjunto de cidades para o usuário. Duas estratégias para recomendação de cidades são desenvolvidas. A primeira utiliza filtragem colaborativa baseada em vizinhança para recomendar cidades, enquanto a segunda utiliza uma técnica de filtragem colaborativa do estado da arte baseada em fatores latentes. A estratégia baseada em vizinhança explora dois tipos de informações distintas: (i) co-ocorrência de cidades visitadas pelos usuários; (ii) avaliações feitas pelos usuários em uma segunda camada - as atrações das cidades, tendo como premissa a robustez quanto à esparsidade dos dados. Na estratégia baseada em fatores latentes, são explorados dois tipos de características: (i) textuais, que envolvem o texto das revisões feitas pelos usuários acerca das atrações das cidades; (ii) geográficas, que utiliza a posição geográfica das cidades como fator decisor para a recomendação. Para as características textuais, são propostas quatro métricas distintas de modo a capturar o valor textual discriminativo das revisões feitas pelos usuários com relação às atrações. Para os atributos geográficos, explora-se a característica intrínseca de distância geográfica entre as cidades, de modo a tratar a proximidade geográfica e a co-ocorrência das cidades em um mesmo problema de otimização. Os métodos propostos foram avaliados em uma base de dados coletada do maior site de turismo do mundo, o TripAdvisor. Os resultados possuem ganhos significativos em precisão quando comparados ao estado-da-arte. Vale ressaltar que as técnicas propostas são flexíveis, podendo ser utilizadas em outros cenários que exploram uma segunda camada de dados textuais ou mesmo que exploram características geográficas dos itens recomendados.

Abstract

Recommender Systems play a key role in the user decision process in Web systems. In tourism, these systems are largely used for recommending hotels, tourist attractions, accommodations, etc. Recently, recommender systems have been used to provide services that can aid users to create their own travel plan. A travel plan consists of several stages, such as: (i) destination choice, (ii) attraction selection, (iii) choosing accommodations, (iv) defining routes, among others. This dissertation focus on the first stage, which consists in recommending a set of cities to the user. Two approaches for cities recommendations are developed. The first uses collaborative filtering based on neighborhood, while the second uses a state-of-the-art collaborative filtering technique based on latent factors. The strategy based on neighborhood exploits two distinct sources of information: (i) co-occurrence of user visited cities; (ii) user's ratings in a second layer – the attractions of cities, which demands techniques that are robust to data sparseness. In the latent factor strategy, two different characteristics are considered: (i) textual, involving user attraction reviews text; (ii) geographic, which uses the cities geographic position as a decision factor to the recommendation. For textual characteristics, we propose four distinct metrics to capture the textual discriminatory power of users reviews in relation to attractions. For geographic attributes, we exploit the intrinsic characteristic of geographic distance between cities, in order to consider geographic closeness and co-occurrence of cities in the same optimization problem. The proposed methods were evaluated in a dataset collected from the largest tourism Web site in the world, TripAdvisor. The results show significant improvements in precision when compared to the state-of-the-art. It is worth mentioning that the proposed techniques based on latent factors are flexible, therefore it can be applied in other scenarios that exploits a textual second layer or even problems that exploit geographic characteristics of recommended items.

Lista de Figuras

Lista de Tabelas

Sumário

Capítulo 1

Introdução

Esta dissertação aborda estratégias de exploração de informações específicas de domínio para melhorar a acurácia em recomendações para turismo. Neste capítulo será apresentada a motivação deste trabalho e será definido o problema a ser enfrentado. Além disso, são apresentados os objetivos e as contribuições deste estudo.

1.1 Motivação

Nos últimos anos, o turismo tem experimentado um crescimento significativo no mundo, em parte por conta da vasta quantidade de informação disponível na Web [Batet et al., 2012]. Essa informação permite aos viajantes um maior conhecimento acerca das possibilidades de viagens, ao mesmo tempo que dificulta sua escolha por conta do número crescente de opções com as quais ele tem que lidar nos processos relacionados à viagem. Além disso, a Web também permite que os negócios de turismo ofereçam serviços online a um custo relativamente baixo [Kim, 2004], o que faz com que esse crescimento se intensifique ainda mais.

Como resultado, recomendações para turismo têm se tornado um problema cada vez mais relevante, pois permitem que os turistas descubram mais facilmente novos destinos, pontos de interesse e/ou rotas de viagem que estejam mais relacionadas às suas preferências, além de permitirem que os sistemas online para turismo alcancem a audiência certa por meio de seus anúncios [Batet et al., 2012]. Os sistemas online para turismo têm atraído a atenção tanto da indústria quanto da academia [Castillo et al., 2008]. Em 2015 a United Nations World Tourism Organization (UNWTO) previu que o turismo internacional irá crescer entre 3% e 4%, com um volume de circulação financeira estimado em 1.5 trilhões de dólares [UNWTO, 2014]. A *eMarketer*, uma agência de

pesquisa de mercado, projeta que o valor das compras de viagens feitas em *smartphones* e *tablets* irá subir de 26 bilhões em 2014 para 65 bilhões em 2018 [eMarketer, 2015].

Da perspectiva da academia, os sistemas de recomendação para turismo têm sido estudados extensivamente [Borrís et al., 2014]. O problema enfrentado por esses sistemas é particularmente complicado por duas razões. A primeira razão é que uma recomendação neste contexto pode ser aplicada em diferentes estágios de um plano de viagem. Os estágios de um plano de viagem podem ser descritos como: (i) escolher destinos, (ii) selecionar atrações turísticas, (iii) escolher acomodações, (iv) decidir rotas, entre outros [Huang & Bian, 2009]. Para que recomendações interessantes possam ser providas, cada um desses estágios deve ter em conta objetivos, desafios e abordagens diferentes, uma vez que precisam capturar características distintas em relação a um mesmo plano de viagens.

A segunda razão consiste no fato de que em sistemas online para turismo, as ações tomadas pelos usuários são muito menos frequentes do que em outros domínios, por conta da natureza do sistema ser relacionada a viagens, algo que é feito, em média, menos de sete vezes por ano por pessoa nos Estados Unidos [USTA, 2014]. Isso difere substancialmente os sistemas de recomendação para turismo de outros, tais como sistemas de recomendação de filmes [Koren, 2008; Miller et al., 2003] ou sistemas de recomendação de músicas [Barrington et al., 2009]. O número de avaliações de itens feitas pelos usuários de sistemas de recomendação para turismo, tende a ser menor do que nos sistemas de outros domínios já largamente explorados na literatura [Garcia et al., 2011]. Por conta disso, modelar precisamente os perfis de usuários, o que é naturalmente necessário para que se alcance recomendações precisas, é uma tarefa mais árdua no turismo online do que em outros domínios [Garcia et al., 2011].

O problema de recomendações de plano de viagens foi estudado em vários trabalhos distintos [Huang & Bian, 2009; Ye et al., 2011a; Yuan et al., 2013]. Na maior parte deles, assume-se que já se possui um destino prefixado e pretende-se recomendar atrações. Em outros pretende-se recomendar conjuntos de cidades para grupos de pessoas ou mesmo planos de visitas inteiros [Gionis et al., 2014; Kurashima et al., 2010]. No entanto, são escarsos os trabalhos feitos com foco na primeira etapa deste problema – que consiste em recomendar as cidades que um usuário deveria visitar dado o seu perfil – e que exploram informações específicas de domínio. Existem trabalhos que abordam esta primeira etapa, porém as técnicas propostas não exploram informações específicas de domínio [Ricci, 2002; Ricci et al., 2006], tais como texto relacionado a revisões de atrações turísticas, a informação de geo-localização ou mesmo categorias de cidades e atrações.

Este trabalho explora informações específicas de domínio com o intuito de endere-

çar a primeira etapa do problema de recomendação de plano de viagens. Uma solução adequada para esta etapa traz duas vantagens claras: (i) permite que os usuários escolham seu destino de viagens mais rapidamente; (ii) aumenta a probabilidade de a escolha feita se adequar ao perfil do usuário, o que pode significar um número maior de usuários satisfeitos com o resultado de suas viagens.

1.2 Definição do Problema

O problema tratado por esta dissertação consiste em recomendar um destino (i.e., cidade) que se adeque às preferências de cada usuário. Assim, dado um conjunto de cidades visitadas por um usuário, a solução proposta deverá identificar, baseado em suas preferências e informações adicionais herdadas pelos itens, as top-N cidades que melhor se adequam aos interesses do respectivo usuário. Essa tarefa está descrita na Figura 1.1.

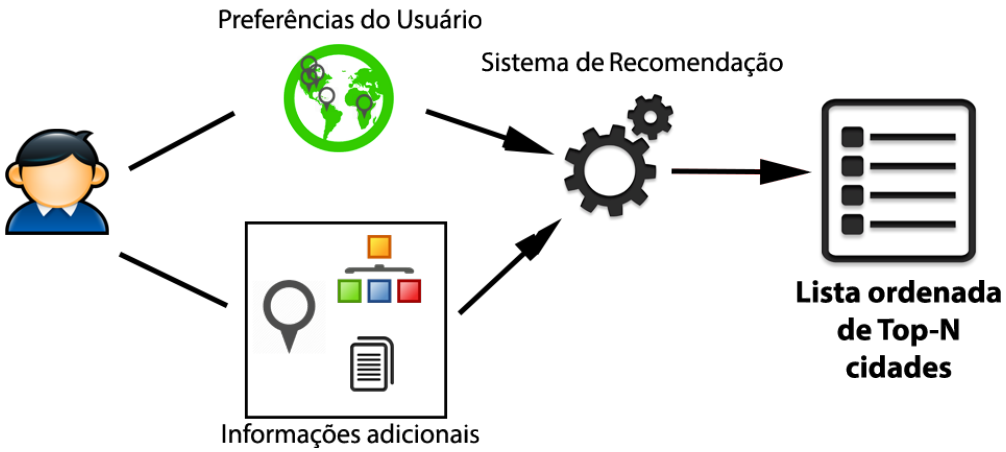


Figura 1.1: Tarefa de recomendação de cidades top-N. A partir das cidades já visitadas pelo usuário e de informações adicionais dos itens, gera-se uma lista ordenada top-N que melhor se adequa aos interesses do usuário.

1.3 Objetivos

O objetivo geral desta dissertação consiste em investigar formas para se fazer recomendações de turismo, utilizando informações adicionais relacionadas ao domínio de turismo. Como objetivos específicos, pretende-se:

- Propor novas estratégias de recomendação de cidades que explorem informações geográficas, informações textuais e que também explorem a hierarquia ci-

dade/atração/texto da atração que pode ser encontrada em sistemas online para turismo. Esta hierarquia apresentada nos sistemas de turismo reflete um aspecto do mundo real, onde cidades possuem atrações que, por sua vez, recebem avaliações (texto) dadas pelos usuários;

- Avaliar experimentalmente os métodos, comparando o desempenho das estratégias propostas entre si e com métodos de recomendação estado-da-arte em uma base de dados coletada de um sistema real.

Dados esses aspectos, este trabalho visa responder as seguintes questões de pesquisa:

- Informações adicionais acerca das cidades podem ser exploradas de modo a se conseguir uma melhor acurácia nas recomendações?
- Informações adicionais de segundo nível de hierarquia (i. e., de atrações) podem ser utilizadas como descritores da cidade a que estão relacionadas?

1.4 Contribuições

As principais contribuições desta dissertação são discutidas a seguir. Parte delas foram publicadas nos seguintes trabalhos: Bidart et al. [2014a,b].

- **Proposição de novas estratégias de recomendação de cidades**: para resolver o problema de recomendação de cidades, foram propostas abordagens que capturam aspectos distintos dos dados: filtragem colaborativa baseada em vizinhança e filtragem colaborativa baseada em fatores latentes. A primeira técnica baseada em vizinhança, explora principalmente a característica hierárquica das entidades que compõem um sistema online para turismo (i. e., cidades/atrações/revisões), de modo a descrever a importância de uma cidade para um usuário por meio de sua presença na comunidade daquele usuário (a comunidade de um usuário é dada pelos k vizinhos mais próximos a ele) e também pelas avaliações dadas pelos usuários da comunidade para as atrações que compõem a cidade. Essa estratégia possui aspectos específicos que permitem que ela seja robusta à escassez de dados de avaliações de atrações. Até onde sabemos, esta é a primeira técnica criada que procura explorar este aspecto hierárquico presente no domínio de turismo com o intuito de melhorar a acurácia das recomendações.

  Para a segunda técnica, baseada em fatores latentes, exploram-se (i) características textuais que envolvem o texto das revisões feitas pelos usuários acerca

das atrações das cidades e (ii) características geográficas, que utiliza a posição geográfica das cidades para se melhorar a recomendação. Para as característics textuais, foram propostas quatro métricas distintas, a saber TF-IDF, wTS, wTF, STAB (veja mais detalhes no Capítulo 3), que procuram capturar o potencial discriminativo do texto das atrações de cada cidade. Note que existe um fator importante que o texto não é diretamente associado às cidades, o que exige adaptações a essas técnicas propostas em outros trabalhos. Para as características geográficas, explorou-se a relação intrínseca de distância geográfica entre as cidades, de modo que a co-ocorrência e a distância entre as cidades fossem atributos tratados em um mesmo problema de otimização. Foi criado um conjunto de métodos (GeoSSLIM) que endereçam este problema e superam métodos do estado-da-arte.

- **Avaliação dos métodos propostos**: os métodos propostos foram avaliados, em termos de precisão de recomendações top-N, em dados reais provenientes do TripAdvisor. Os resultados foram comparados a duas técnicas estado-da-arte: WRMF [Koren, 2008] e SLIM [Ning & Karypis, 2011]. Os resultados indicam que nossa melhor técnica, o GeoSSLIM, supera os modelos de referência SLIM por 9.06% (@5) e WRMF por 13.32%(@5) em termos de precisão média. É importante ressaltar que as técnicas propostas são flexíveis, podendo ser utilizadas em outros cenários que exploram uma segunda camada de dados textuais ou mesmo que exploram características geográficas dos itens recomendados.

1.5 Organização

O restante deste trabalho está organizado como segue. O Capítulo 2 introduz conceitos básicos e discute os trabalhos relacionados, enquanto o Capítulo 3 se aprofunda no problema de recomendação de cidades e descreve as técnicas que foram desenvolvidas para tal tarefa. O Capítulo 4 descreve a base de dados utilizada, juntamente com a metodologia de avaliação aplicada. Ainda no Capítulo 4 são apresentados os principais resultados alcançados a partir das técnicas propostas. Por fim, as conclusões e direções para trabalhos futuros são discutidas no Capítulo 5.

Capítulo 2

Contextualização e Trabalhos Relacionados

No Capítulo 1 foram apresentadas as razões que motivaram este trabalho, foi definido o problema a ser tratado e os objetivos desta dissertação, juntamente com suas contribuições. Este capítulo desenvolve um estudo exploratório da literatura em questão, tanto no que concerne às bases teóricas quanto dos trabalhos práticos que já foram desenvolvidos e estão relacionados à proposta desta dissertação. Inicia-se apresentando os sistemas de recomendação de um modo geral, seguidos por aqueles que utilizam informações específicas de domínio e, posteriormente, abordando os sistemas que são voltados especificamente para o domínio de turismo.

2.1 Sistemas de Recomendação

Sistemas de recomendação são serviços que provêem sugestões para que itens sejam utilizados por usuários. As sugestões providas têm o objetivo de dar suporte aos usuários em vários processos de tomada de decisão, tais como que itens comprar, que músicas ouvir ou que notícias ler [Ricci et al., 2011]. Os sistemas de recomendação podem ser divididos em dois tipos principais: baseados em conteúdo e baseados em filtragem colaborativa, os quais se diferem principalmente no que concerne ao ponto focal da técnica de recomendação. Para o primeiro, o perfil de um usuário é descrito a partir dos itens que consome, enquanto o segundo considera a semelhança com outros usuários como fator primordial para se definir o perfil de um usuário.

2.1.1 Tipos de Sistemas de Recomendação

Sistemas que implementam uma abordagem **baseada em conteúdo** analisam um conjunto de documentos e/ou descrições de itens previamente avaliados pelo usuário de modo a construir um modelo ou perfil dos seus interesses, a partir das características dos itens avaliados por ele [Ricci et al., 2011]. O perfil é basicamente uma representação estruturada dos interesses do usuário, adotada para recomendar novos itens. O processo de recomendação consiste basicamente em correlacionar os atributos do perfil do usuário com os atributos de um item, de modo que o resultado da recomendação torna-se uma inferência do nível de interesse do usuário naquele item. Esta técnica pode ser usada, por exemplo, para filtrar resultados de uma busca onde se decide se um usuário está interessado em uma página Web específica ou não. A Figura 2.1 representa um modelo de recomendação baseado em conteúdo. Note que é a similaridade entre um item que o usuário já gostou no passado com um item que ele ainda não conhece o fator definidor da recomendação.

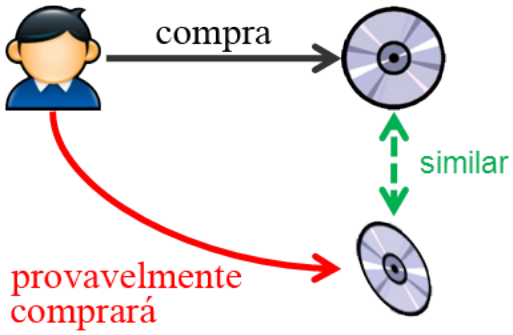


Figura 2.1: Recomendação baseada em conteúdo. Na recomendação baseada em conteúdo apenas a similaridade entre os itens que que o usuário não conhece e os itens que ele já gostou é levada em conta para se realizar a recomendação.
Fonte: http://horicky.blogspot.com.br

Ao contrário dos sistemas baseados em conteúdo, os sistemas que utilizam uma abordagem **baseada em filtragem colaborativa** baseiam suas predições nas avaliações ou comportamento dos usuários, ou seja, centralizam sua abordagem na semelhança de perfil entre usuários, não na semelhança entre itens. A hipótese fundamental dessa abordagem é que as opiniões de usuários podem ser selecionadas e agregadas de modo a prover predições razoáveis das preferências dos mesmos. Intuitivamente, a abordagem assume que, se os usuários concordam acerca da qualidade ou relevância de alguns itens, então eles irão concordar também quando forem analisar outros itens [Ekstrand et al., 2011]. Por exemplo, se um grupo de usuários gosta das mesmas coisas que uma usuária hipotética "Maria", então é provável que a usuária "Maria" irá

gostar dos itens que eles gostam mas ela ainda não conhece. A abordagem baseada em filtragem colaborativa tem obtido muito sucesso tanto em pesquisa como em aplicações práticas [Sarwar et al., 2001], por isso ela tem sido a estratégia mais utilizada nos sistemas de recomendação da atualidade. A Figura 2.2 representa um modelo de recomendação baseada em filtragem colaborativa. Note que é a similaridade entre usuários que determina os itens que serão recomendados.

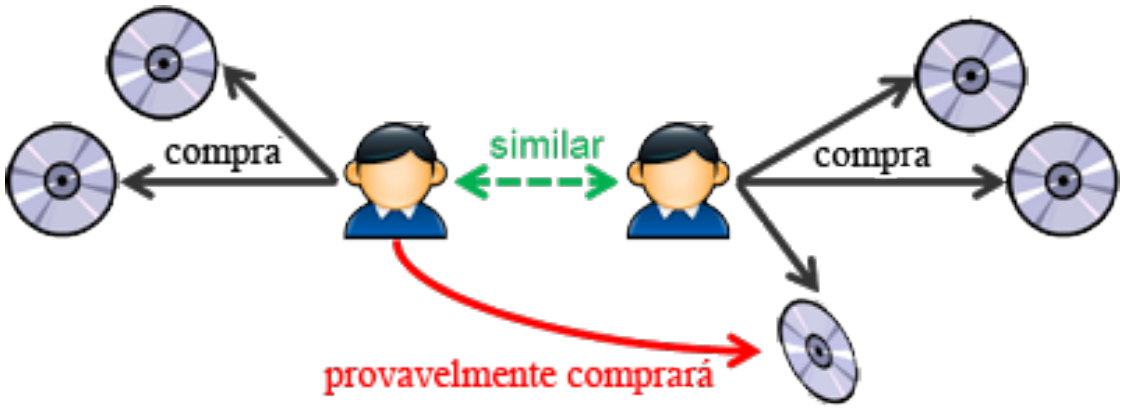


Figura 2.2: Recomendação baseada em filtragem colaborativa. Na recomendação baseada em filtragem colaborativa a similaridade entre perfis de usuários é o fator determinante para se realizar a recomendação. Na imagem, a semelhança entre o consumo dos dois usuários faz com que um item que foi consumido apenas por um deles seja recomendado para o outro.
Fonte: http://dataconomy.com

2.1.2 Tarefas de Recomendação

Um sistema de recomendação pode ser utilizado ou para predizer se um determinado usuário irá gostar de um item em particular (predição de nota) ou para identificar um conjunto de N itens que serão de interesse de um certo usuário (recomendação top-N) [Deshpande & Karypis, 2004].

A tarefa de predição de nota consiste em predizer as notas que os usuários darão para itens ainda não avaliados. Essa tarefa tem seu resultado comumente avaliado pela mensuração dos erros cometidos nas predições, utilizando-se medidas como MAPE (Mean Absolute Percentage Error) e RMSE (Root Mean Squared Error) [Hyndman & Koehler, 2006]. A Figura 2.3 sumariza um exemplo do que se quer realizar em uma tarefa de predição. Nela pode ser vista uma tabela onde as linhas são usuários e as colunas os itens. Os usuários deram notas para alguns itens, mas não todos. A tarefa

consiste em prever quais seriam os valores dados pelos usuários para itens ainda não avaliados, utilizando como base as notas já existentes.

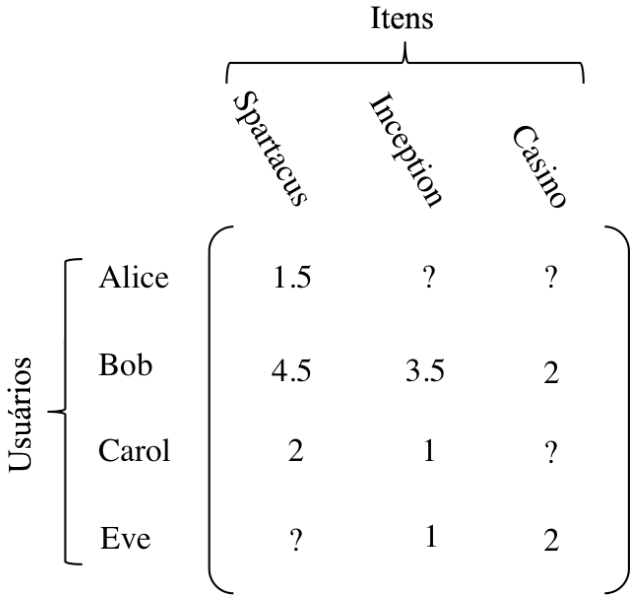


Figura 2.3: Tarefa de predição de notas. O objetivo é encontrar o melhor valor que prevê a nota que cada usuário daria para itens que ele ainda não avaliou

A tarefa de recomendação top-N, por sua vez, consiste em identificar N itens que serão de interesse de um determinado usuário. Neste caso, a nota dada pelo usuário para cada item não é o mais importante, mas sim a ordem em que cada item aparece no lista ordenada de itens de cada usuário. Deste modo, métricas de acurácia, tais como precisão e NDCG (Normalized Discounted Cumulative Gain), tornam-se mais adequadas para avaliação desta tarefa do que métricas de erro, por serem de mais natural interpretação [Cremonesi et al., 2010]. Uma descrição melhor desses sistemas pode ser vista na Figura 2.4, que demonstra à direita o aspecto principal dos sistemas top-N, que consiste em recomendar uma lista ordenada de itens personalizada para cada usuário. A Figura 2.4 assemelha-se à Figura 2.3. Porém, elas se diferenciam pelo fato de o objetivo da recomendação top-N não estar relacionado à predição das notas, mas sim na predição de um ranking de N elementos para cada usuário. O foco deste trabalho é a tarefa de recomendação top-N.

2.1.3 Tipos de Feedback

Em sistemas de recomendação, as preferências dos usuários são inferidas levando-se em consideração *feedback* direto do usuário, que pode ser nas formas ímplicita ou

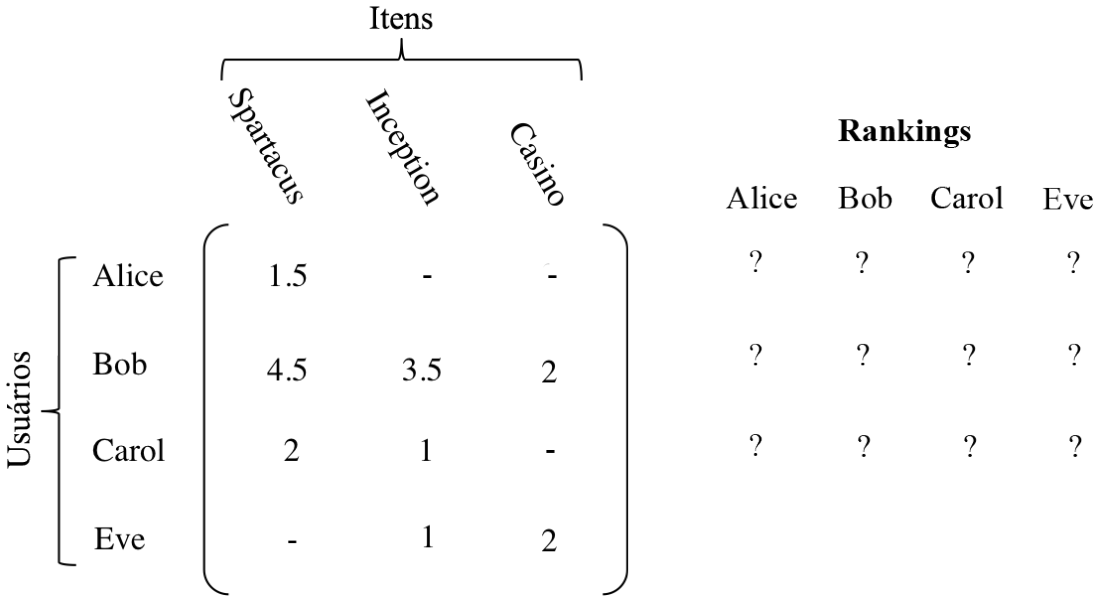


Figura 2.4: Tarefa de recomendação top-*N*. O objetivo é, para cada usuário, encontrar uma lista de itens ordenados por relevância.

explícita [Parra et al., 2011]. O *feedback* implícito é obtido ao se mensurar a interação do usuário com diferentes itens. Pode-se utilizar sinais indiretos, tais como a quantidade de vezes que uma música foi executada ou o número de clicks em uma página Web [Oard et al., 1998]. Este tipo de dados é obtido sem incorrer em qualquer sobrecarga para o usuário, já que surge a partir do uso natural dos sistemas. Por outro lado, o *feedback* explícito é obtido pela consulta direta ao usuário, que é normalmente apresentado a uma escala onde ele quantifica o quanto gosta de determinado item. Naturalmente, o *feedback* explícito é uma forma mais robusta de extrair a preferência do usuário, já que ele está apontando diretamente aquilo que gosta, removendo a necessidade de inferência indireta [Parra et al., 2011]. Nesta dissertação há um interesse particular em estratégias que exploram *feedback* implícito, já que que grande parte das informações específicas de domínio que estão disponíveis possuem esta característica.

2.1.4 Métodos de Recomendação

Segundo Adomavicius & Tuzhilin [2005], métodos baseados em filtragem colaborativa, que são foco principal desta dissertação, podem ser agrupados em duas classes gerais: os métodos baseados em vizinhança e baseados em fatores latentes. Nos métodos baseados em vizinhança – que também são chamados de baseados em memória [Breese et al., 1998] ou baseados em heurística [Adomavicius & Tuzhilin, 2005] – as notas usuário-item armazenadas no sistema são diretamente utilizadas para predizer

notas de novos itens. A recomendação baseada em vizinhança pode ser feita por meio de duas abordagens: recomendação centrada no usuário ou recomendação centrada no item. Sistemas centrados no usuário avaliam o interesse de um usuário u por um item i usando as notas dadas para os itens por outros usuários possuem padrões de avaliação similares a u (por isto são chamados de vizinhos). Por outro lado, nos sistemas centrados no item, a predição da nota de um usuário u para um item i é baseada nas notas que um usuário u forneceu para itens similares a i [Ricci et al., 2011]. Nesta abordagem, dois itens são similares se vários usuários do sistema avaliaram estes itens de maneira semelhante.

Em contraste com os métodos baseados em vizinhança, que usam notas diretamente em sua predição, as abordagens usando fatores latentes usam estas notas apenas para aprender um modelo preditivo. A ideia central desse método é modelar as interações usuário-item como fatores que representam características latentes dos usuários e itens no sistema, tais como classe de preferência de usuários e classe de categoria de itens [Ricci et al., 2011]. Este modelo é então treinado usando os dados disponíveis e depois utilizado para predizer as notas de usuários para novos itens. Abordagens baseadas em fatores latentes são numerosas e incluem Support Vector Machines (SVM) [Grčar et al., 2006], Singular Value Decomposition (SVD) [Koren, 2008; Paterek, 2007] e Sparse LInear Methods (SLIM) [Ning & Karypis, 2011].

Os métodos baseados em fatores latentes são considerados estado-da-arte para o problema de recomendação top-N [Kabbur et al., 2013]. Dois desses métodos do estado-da-arte são SLIM [Ning & Karypis, 2011] e SSLIM [Ning & Karypis, 2012], que são particularmente estudados nesta dissertação. O SLIM é um método que foca em fazer recomendações top-N de alta qualidade e rapidamente. Este método aprende, resolvendo um problema de otimização com regularização, uma matriz de coeficientes esparsa para os itens no sistema somente a partir dos perfis de compra/avaliação. O SLIM resolve o problema de otimização a partir de uma solução linear que é também paralelizável, isto faz com que seja um método eficiente. Além disso, o SLIM consiste no estado-da-arte no que concerne à acurácia nas recomendações [Ning & Karypis, 2011]. Desde modo, este método endereça a demanda por sistemas recomendadores eficientes e de alta qualidade na tarefa de recomendação top-N de maneira concorrente, sendo assim adequado para aplicações que necessitam de recomendadores que são executados em tempo real. Já o SSLIM consiste basicamente em uma extensão do SLIM onde se agregam informações adicionais ao problema de otimização endereçado pelo SLIM. O SSLIM é especialmente aplicável em problemas onde há informações adicionais acerca de itens, uma vez que ele é capaz de utilizar estas informações adicionais de modo a alcançar recomendações com maior acurácia do que o SLIM [Ning & Karypis, 2012].

2.2 Recomendação Usando Informações Específicas de Domínio

O uso de informações específicas de domínio para recomendação consiste em um tópico bem estudado na literatura. Este tipo de algoritmo pode ser dividido em três classes distintas: (i) os que exploram atributos de entidades (usuários e itens), tais como sexo, idade, localização ou palavras que descrevem os itens; (ii) algoritmos que exploram a dinâmica temporal dos sistemas; (iii) os que exploram as redes sociais entre usuários e outras formas de informação de redes [Gunasekar, 2012]. O primeiro tipo de algoritmo é a forma mais comum de informação adicional disponível e consiste no foco deste trabalho.

Diversas técnicas foram criadas com o intuito de explorar informação adicional de entidades. Uma abordagem simples para solução deste problema consiste em combinar um modelo de filtragem colaborativa com outro baseado em conteúdo [Claypool et al., 1999]. No entanto, esta solução tende a ser muito custosa em termos de memória e computação, uma vez que é preciso que sejam mantidos dois modelos no lugar de apenas um. Um exemplo de uma solução que tenta combinar esses dois modelos foi desenvolvida por Balabanović & Shoham [1997], e utiliza basicamente modelos de vizinhança para fazer a recomendação.

Existem publicações na literatura que são baseadas em *filterbots*. *Filterbots* consistem em incluir, no modelo de filtragem colaborativa, agentes cujas notas são diretamente baseadas no conteúdo, os quais são chamados de *filterbots*. Modelos utilizando esta técnica podem ser encontrados em [Pazzani, 1999; Good et al., 1999].

Também existem as técnicas que são baseadas em fatores latentes e são consideradas as implementações de maior sucesso [Gunasekar, 2012]. Algumas delas exploram modelos de regressão [Agarwal & Chen, 2009], outras que exploram modelos probabilísticos como LDA [Agarwal & Chen, 2010] e redes bayesianas [Porteous et al., 2010], além de modelos baseados em fatoração de matrizes [Shan & Banerjee, 2010; Ning & Karypis, 2012].

As informações específicas de domínio em conjunto com os métodos para explorá-las possuem grande valor por permitirem que se correlacionem aspectos específicos de cada domínio de modo a se alcançar recomendações que se relacionam melhor às preferências dos usuários. Este tipo de sistemas tem sido largamente explorado nos mais diversos domínios [Gunasekar, 2012]. Alguns exemplos de domínios são recomendação de músicas [van den Oord et al., 2013; Wang & Wang, 2014; Ostuni et al., 2013], de filmes [Ostuni et al., 2013; Ning & Karypis, 2012], imagens [Xie et al., 2015] ou mesmo

de comunidades científicas [Fang & Si, 2011]. No contexto de turismo há pouca exploração de técnicas que utilizam informações específicas de domínio para recomendação. Pode-se destacar a tarefa de recomendação de atrações, a qual é endereçada por vários trabalhos [Kurashima et al., 2010; Ye et al., 2011b].

2.3 Recomendação para Turismo

Sistemas de recomendação têm sido aplicados em diferentes domínios, sendo que neste trabalho o foco é aplicações em turismo. Recentemente, redes sociais online de viagens têm modificado o modo com que os turistas planejam suas viagens, pois os turistas têm utilizado cada vez mais informações destes Web *sites* para tomar suas decisões de viagem [Ayeh et al., 2013]. Estes Web *sites* permitem que os usuários consumam e forneçam revisões de hotéis, restaurantes ou de atrações (p.ex., museus e parques). Alguns exemplos destes Web *sites* são TravBuddy.com, TravellersPoint.com e TripAdvisor.com. Este último é considerado como a maior comunidade de viagens na Web [Tri, 2015a].

Borrís et al. [2014] desenvolveram uma pesquisa abrangente e aprofundada acerca dos sistemas de recomendação para turismo, analisando diferentes linhas de pesquisa desde 2008. O artigo publicado por eles, provê um exame detalhado e recente acerca do campo de pesquisa, considerando diferentes tipos de interfaces, algoritmos de recomendação e funcionalidades oferecidas por estes sistemas, além de abordar o uso de técnicas de inteligência artificial neste contexto.

Os sistemas de recomendação para turismo são capazes de fornecer sugestões relevantes para turistas sempre que visitam lugares desconhecidos. Isto é feito por meio de ferramentas de suporte para tornar o processo de decidir o que fazer mais simples. Como apontado por Borrís et al. [2014], a maioria das abordagens sugere atrações em um destino de acordo com as preferências dos usuários. Estas sugestões podem ser apresentadas para o usuário em uma lista ordenda pela importância ou podem ser integradas a um plano programado.

As abordagens mais populares para recomendação são as baseadas em filtragem colaborativa. No entanto, no contexto de turismo, existem ainda algumas abordagens que aplicam somente métodos baseados em conteúdo [Ricci, 2002].

Existem muitas pesquisas sobre sistemas de recomendação para turismo, as quais podem ser classificadas em quatro grandes categorias, dependendo do serviço que oferecem: (i) sugestão de um destino e construção de um pacote inteiro de turismo [Seidel et al., 2010; Yu & Chang, 2009; Lorenzi et al., 2011; Saso & Biljana, 2012]; (ii) reco-

mendação de atrações em um destino específico [Yang & Hwang, 2013; Garcia et al., 2013b; Savir et al., 2013; Braunhofer et al., 2013]; (iii) concepção de uma programação de viagem de múltiplos dias [Batet et al., 2012; Vansteenwegen & Souffriau, 2010; Lucas et al., 2013] e (iv) recursos sociais [Ceccaroni et al., 2009; Garcia et al., 2013a; García-Crespo et al., 2011]. Todas essas categorias possuem características específicas e podem ser abordadas com técnicas específicas para cada uma.

Este trabalho foca em construir técnicas de recomendação que exploram informação específica do domínio de turismo, direcionadas para a primeira etapa deste processo: recomendação de destinos. Neste contexto, até onde sabemos, não há trabalhos que exploram três aspectos que pretende-se explorar nesta dissertação: (i) conteúdo gerado pelo usuário (texto) em uma segunda camada (atrações), que é comum em vários sistemas online para turismo; (ii) posição geográfica das cidades em conjunto com as preferências dos usuários em uma mesma abordagem; (iii) aspectos não textuais da segunda camada (atrações) como característica para melhorar a recomendação.

2.4 Sumário

Neste capítulo contextualizamos o trabalho em relação aos diversos aspectos da literatura de recomendação para a exploração de informações específicas de domínio para a realização de recomendações top-N no contexto de turismo. Além disso, demonstramos a vasta literatura que compõe cada um dos tópicos envolvidos dando ênfase aos modelos de recomendação, aos trabalhos que exploram informações específicas de domínio e às aplicações de recomendação para turismo. No Capítulo 3 utilizaremos o contexto aqui apresentado como base para descrever, em detalhes, os métodos de recomendação para turismo desenvolvidos nesta dissertação.

Capítulo 3

Recomendação de Cidades para Turismo

No capítulo anterior foram descritos os aspectos mais relevantes para a compreensão deste trabalho e os trabalhos relacionados. Neste capítulo serão descritas as três estratégias de recomendação de cidades propostas por esta dissertação.

3.1 Formalização do Problema

Seguindo a notação proposta por Ning & Karypis [2012] para cenários de *feedback implícito* e considerando informação adicional associada a itens referenciada aqui como *informação adicional*, problema tratado nesta dissertação é definido como:
Seja U o conjunto de todos os usuários e D o conjunto de todos os destinos (itens). As estratégias de recomendação de cidades aqui descritas exploram como principais fontes de dados:

1. A matriz de *feedback implícito* $A \subseteq U \times D$, onde $a_{u,d} = 1$ se d é relevante para o usuário u (i.e., usuário u visitou o destino d). Denota-se o tamanho do conjunto de usuários e de itens por m e n, respectivamente. A partir de A, define-se a_i^T, isto é a i-ésima linha de A, como sendo o perfil de visitação do usuário i em relação a todos os destinos. A j-ésima coluna de A representa o perfil de visitação do destino j em relação a todos os usuários;

2. Para cada item d, define-se um vetor de *informação adicional* f_d, onde $f_d = 1$ quando a informação adicional está presente para aquele item. A informação adicional para todos os destinos é representada por uma matriz $B_{k \times n}$, onde k

representa o tamanho da *informação adicional*. A j-ésima coluna de B representa o vetor de informação adicional do destino j (i.e., f_j);

3. Define-se também *atrações* como locais que podem ser visitados em uma cidade (p.ex., museu, parque, etc.) e *revisões* como conteúdo textual gerado pelos usuários acerca de uma *atração*. Note que essas entidades são inter-relacionadas em uma hierarquia onde *usuários* visitam *destinos* que, por sua vez, possuem *atrações* as quais são visitadas pelos *usuários*. Aqui considera-se que as *revisões* são associadas às atrações ao invés das cidades, como ocorre em várias aplicações reais (p.ex., TripAvisor [1], Yelp [2] e Yahoo! Travel [3]). Uma ilustração desta hierarquia pode ser vista na Figura 3.1.

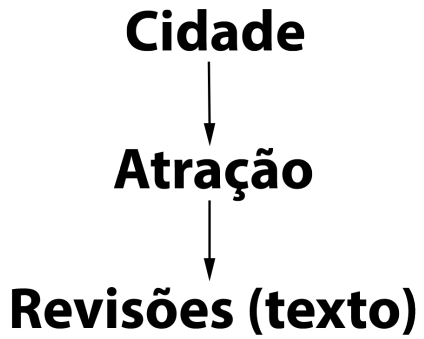


Figura 3.1: Hierarquia de entidades em um sistema online para turismo. Atrações estão dentro de cidades e revisões são feitas pelos usuários acerca das atrações.

O problema tratado por esta dissertação consiste em recomendar um destino (i.e., uma cidade) que se adeque às preferências de cada usuário. Assim, dado um conjunto de cidades visitadas por um usuário, a solução proposta deverá identificar, baseado em suas preferências e informações adicionais herdadas pelos itens, as top-N cidades que melhor se adequam aos interesses do respectivo usuário alvo da recomendação.

3.2 Estratégias de Recomendação

Como descrito no Capítulo 1, o problema de recomendação para turismo pode ser segmentado em quatro etapas, sendo a primeira delas recomendação de cidades. Este trabalho foca especialmente no uso de informações específicas de domínio para

[1] www.tripadvisor.com
[2] www.yelp.com
[3] travel.yahoo.com

endereçar essa primeira etapa, que conta com baixa quantidade de estudos direcionados a isto [Borrís et al., 2014]. São propostas três estratégias diferentes para abordar o problema, uma baseada em vizinhança, outra em fatores latentes e a terceira uma alternativa para união das duas anteriores em uma. Como as estratégias de vizinhança e de fatores latentes utilizam modelos distintos, são apresentados dois tipos de modelagens para o problema. Para tanto, tem-se uma extensão da notação definida na Seção 3.1:

- **Estratégia Baseada em Vizinhança:** dado o conjunto de usuários U, a estratégia é capaz de gerar uma matriz $\mathbf{T} \in \mathbb{R}^{nxn}$ representando as similaridades entre os usuários, assimilando pesos para pares específicos de usuários, de modo a gerar correlações entre cada um eles. A partir da matriz T, a estratégia deve ser capaz de encontrar, para cada usuário u, um grupo G_u de usuários correlatos que melhor descrevem u (i.e., mais similares a u) – uma tarefa similar à de detecção de comunidades. A partir de G_u, pode-se gerar o grupo de cidades visitadas por todos os usuários na comunidade de u, que denotaremos C_{G_u}. A partir de A, T, G_u, C_{G_u} e as informações adicionais presentes para cada item (B), a estratégia deve ser capaz de gerar uma lista ordenada de cidades para cada usuário;

- **Estratégia Baseada em Fatores Latentes:** dadas as matrizes A e B, o objetivo é encontrar uma matriz W, que representa os fatores latentes encontrados a partir das correlações de visitas entre usuários juntamente com as informações adicionais dos itens. Esta matriz W, por sua vez, é utilizada para se ordenar as cidades a serem recomendadas para cada usuário.

  As duas abordagens, apesar de possuírem o mesmo fim, capturam diferentes características dos usuários e dos itens. Um resumo das duas abordagens pode ser visto na Tabela 3.1. Nas próximas seções deste capítulo as estratégias serão vistas em detalhes.

3.3 Recomendação de Cidades Baseada em Vizinhança

A solução proposta para endereçar o problema de recomendação de cidades por meio de uma estratégia de vizinhança foi chamada de ReCWEE – ***Re****commendation using* ***C****ommunities and* ***W****ithout* ***E****xplicit* ***E****valuations*. A hipótese por trás desta solução concentra-se no aspecto de que usuários que visitaram um grande número de

	Vizinhança	**Fatores Latentes**
Entrada	A: matriz de feedback implícito B: matriz de informação adicional T: matriz de correlação entre usuários G_u: comunidade de cada usuário C_{G_u}: grupo de cidades visitadas pelos usuários da comunidade de u	A: matriz de feedback implícito B: matriz de informação adicional
Saída	Lista ordenada de cidades por usuário	

Tabela 3.1: Descrição do problema de recomendação de cidades em termos de entrada e saída de dados.

cidades em comum com o usuário alvo u têm uma alta probabilidade de visitar outras cidades que u gostaria de visitar (filtragem colaborativa). Em outras palavras, as melhores cidades candidatas para serem recomendadas para u estão entre aquelas visitadas por usuários com interesses similares.

O projeto do ReCWEE considera que não existem avaliações explícitas para as cidades na coleção de dados e que as cidades que foram visitadas por cada usuário u estão disponíveis (a notação utilizada para este conjunto é a_u, conforme Seção 1.2). Este é o caso de sistemas como, por exemplo, o TripAdvisor, que é considerado o maior *Web site* de turismo do mundo, segundo a comScore[4]. O algoritmo usa uma abordagem de recomendação top-N, levando em conta somente o grupo de cidades $C_{g_u} - a_u$, que são as cidades visitadas pela comunidade do usuário u removendo-se aquelas que ele já visitou, visto que deseja-se fazer recomendações apenas de cidades ainda não visitadas pelo usuário alvo.

O ReCWEE é composto por três módulos principais, que são executados independentemente e em série, como ilustrado na Figura 3.2. As próximas subseções detalham cada etapa da solução proposta. A seção atual será finalizada colocando as etapas juntas para construir a estratégia de recomendação baseada em vizinhança, proposta por este trabalho.

3.3.1 Geração do Grafo

Em nosso contexto, a primeira etapa no processo de geração do grafo consiste em construir uma rede vinculando os usuários. Ou, em outras palavras, como podemos inferir similaridade entre pares de usuários? Dado que foi assumido que não existem avaliações explícitas dos usuários acerca das cidades, utiliza-se o conjunto a_u de cidades previamente visitadas pelo usuário u como evidência dos interesses de u. Especifica-

[4]http://www.comscore.com.

Figura 3.2: Visão geral do ReCWEE. Etapa 1: um grafo de usuários é criado. Etapa 2: são detectadas comunidades no grafo. Etapa 3: uma lista ordenada de cidades é gerada e as top-N cidades são recomendadas para o usuário.

mente, a similaridade entre dois usuários u_1 e u_2 é computada a partir da similaridade entre o conjunto de cidades visitadas por eles, a_{u1} e a_{u2}, dada pelo coeficiente de Jaccard Tan et al. [2006]:

$$J(a_{u1}, a_{u2}) = \frac{|a_{u1} \cap a_{u2}|}{|a_{u1} \cup a_{u2}|} \tag{3.1}$$

Note que o coeficiente de Jaccard é calculado entre cada par de usuários, assim preenchendo completamente a matriz T (ver definição na Seção 3.1). Foram feitos experimentos com outras medidas de similaridade (p. ex., similaridade de cossenos), mas o resultado produzido pelo coeficiente de Jaccard foi no mínimo tão bom quanto o obtido por outras métricas. Assim, optou-se por utilizar Jaccard, uma vez que além de seu bom desempenho possui fácil interpretação.

Define-se então um limiar τ de similaridade mínima a ser considerada para que as arestas entre os usuários sejam criadas no grafo. Usuários que se tornarem desconectados de todos os outros após a adição de ligações (i.e., nós isolados no grafo) também serão removidos do grafo, uma vez que não pertencem a nenhuma comunidade.

3.3.2 Detecção de Comunidades

Nesta etapa é utilizado o algoritmo k-Nearest Neighbor (k-NN) para inferir a comunidade dado um usuário u. Este algoritmo simplesmente seleciona os k usuários mais similares a u como sendo sua comunidade. Embora existam outros algoritmos de detecção de comunidades que poderiam ser aplicados nesse passo [Fortunato, 2010; Herlocker et al., 2002], o k-NN foi escolhido por ser uma abordagem direta e escalável. O valor de k foi configurado para 20, com sugerido por Herlocker et al. [2002], o que significa que os 20 usuários mais similares ao usuário u serão selecionados como sendo sua comunidade. Note que o usuário u_2 pode pertencer à comunidade de u_1, enquanto

u_1 pode *não* pertencer à comunidade de u_2.

As comunidades geradas pelo k-NN desempenham um papel chave na próxima etapa do método (ordenação), uma vez que o conjunto de cidades candidatas a serem recomendadas para o usuário u, serão extraídas da comunidade de usuários associadas a ele. Ao reduzir o conjunto de vizinhos para os top-20 mais similares, reduz-se indiretamente o espaço de busca das cidades a serem recomendadas para u.

3.3.3 Ordenação de Cidades Candidatas

Nesta etapa final, as cidades que são candidatas a serem recomendadas para o usuário alvo u serão ordenadas de acordo com sua *utilidade* para u. As cidades candidatas são extraídas da comunidade de u, ou seja, as cidades que foram visitadas por usuários que fazem parte da comunidade de u (G_u), excluindo-se aquelas já visitadas por u. Essa seleção significa que, ao final, as cidades candidatas serão dadas pelo conjunto $C_{G_u} - a_u$. A utilidade de cada cidade para u é estimada por uma função de pontuação. Dada esta função de pontuação, as cidades candidatas são simplesmente ordenadas em ordem decrescente de pontuação.

Uma contribuição chave deste trabalho consiste em propor uma função de pontuação que usa informação de duas camadas – *cidades* e *atrações*. Duas funções para explorar a característica hierárquica do domínio de turismo foram desenvolvidas, uma computacionalmente mais simples e menos robusta à escassez de dados e outra mais robusta, porém, computacionalmente mais cara.

A primeira atribui uma pontuação para uma cidade c candidata à recomendação para o usuário alvo u de acordo com o número de usuários na comunidade de u que visitaram c no passado, ao mesmo tempo que pondera este valor utilizando a média de avaliações de todas as atrações relacionadas à cidade c. Isso significa que a pontuação de c corresponde à sua popularidade na comunidade de u (G_u), ponderando cada cidade pela opinião dos usuários em geral acerca de suas atrações. Essa função de pontuação é definida como:

$$Score_1(c, u) = Popularity(G_u, c) \times MeanAttractionEvaluations(c) \tag{3.2}$$

onde $Popularity(G_u, c)$ é a fração de usuários em G_u que visitaram c no passado.

A segunda função proposta leva em conta não somente a popularidade de uma cidade candidata em G_u, mas também como cada usuário em G_u avaliou cada uma das

atrações contidas na cidade. A equação dessa função é definida como:

$$Score_2(c,u) = Popularity(G_u,c) \times \sum_{u_i \in G_u} \frac{\frac{1}{|Attr_c|} \sum_{a \in Attr_c}^{n} \frac{Eval(a,u_i)}{Bias(u_i)}}{WithRating(G_u,c)} \tag{3.3}$$

onde $Attr_c$ é o conjunto de atrações na cidade c, $Eval(a,u_i)$ é a avaliação que o usuário u_i deu para a atração a em uma escala de 1 a 5 (note que, embora seja uma avaliação explícita, é uma avaliação de atração não de cidade), e $Bias(u_i)$ é a média de todas as notas dadas pelo usuário u_i para atrações. As avaliações de u_i são normalizadas por este fator de modo a capturar qualquer tipo de viés que u_i possa ter em avaliar os itens com notas mais altas ou mais baixas e assim poder agrupar com mais precisão as opiniões de usuários diferentes em G_u acerca de cada atração. O somatório mais interno agrega as opiniões dos vizinhos de u_i com respeito a todas as atrações da cidade c. O valor dessa agregação é então dividido pelo número de atrações em c de modo a alcançar uma média das opiniões de u_i acerca da cidade c, e assim evitar o favorecimento de cidades com um grande número de atrações. Finalmente, leva-se em conta a média das opiniões de todos os vizinhos que avaliaram ao menos uma atração na cidade (o somatório mais externo, onde $WithRating(G_u,c)$ informa o número de usuários em G_u que avaliaram ao menos uma atração na cidade c) e multiplica-se esse valor pela popularidade da cidade no grupo.

Naturalmente as avaliações de atrações são tipicamente esparsas (como observado em nossa coleção de dados, ver Capítulo 4). Em particular, existem poucas avaliações sobre atrações de cidades nas comunidades do usuário alvo, ou mesmo na coleção de dados como um todo. Assim, foi proposta uma adaptação cuidadosa da função $Score_2$ modificando sua aplicação dependendo da quantidade de dados disponível. O problema principal relaciona-se ao conjunto de usuários G_u na Equação 3.3. Foram previstos três cenários:

1. Caso existam avaliações para atrações da cidade dentro da comunidade do usuário: são utilizadas as opiniões dos usuários nesta comunidade, conforme a Equação 3.3;

2. Caso não existam avaliações para as atrações da cidade dentro da comunidade do usuário, mas existem avaliações de outros usuários (que não fazem parte da comunidade): deve-se substituir G_u na Equação 3.3 por todos os usuários do grafo que já avaliaram alguma atração naquela cidade;

3. Caso não existam avaliações de atrações da cidade: a função de pontuação reduz-se a somente avaliar a popularidade (função *Popularity*) da cidade em G_u.

3.3.4 Unindo as Etapas: ReCWEE

O pseudocódigo do ReCWEE é apresentado pelo Algoritmo 1. Ele recebe como entrada o usuário alvo u bem como o conjunto de todos os usuários U. O primeiro passo é gerar um grafo (linha 25) por meio da função *GraphGeneration* (linhas $1-12$), a qual cria um grafo completo entre os usuários sendo que as arestas têm seu peso dado pelo coeficiente de Jaccard calculado entre as cidades visitadas por cada dos usuários. A seguir, eliminam-se as arestas fracas e nós isolados, de acordo com o limiar de similaridade τ previamente definido.

Depois disso, o algoritmo k-NN é utilizado para gerar a comunidade para o usuário alvo u (linha 26). Finalmente, a função *Ranking* é chamada para produzir uma lista ordenada de cidades candidatas para o usuário u. As top-N cidades são então recomendadas (linha 27). Note que a primeira etapa, que é a mais custosa, não necessita ser feita em tempo real, podendo ser realizada na fase de treinamento, assim como a etapa de detecção de comunidades. Apenas a geração da lista ordenada de cidades precisa ser realizada em tempo de recomendação.

A função *Ranking* (linhas $14-21$) recebe o usuário u e sua comunidade como parâmetros e retorna uma lista ordenada de cidades para ser recomendada para o usuário alvo. Ela primeiro determina as cidades candidatas, utilizando para isso as cidades que foram visitadas pelos usuários da comunidade de u e removendo aquelas que u já visitou. Cada cidade recebe uma pontuação (linha 17), utilizando a Equação 3.2 ou a Equação 3.3. A lista de cidades candidatas é então ordenada pela pontuação em ordem reversa e então retornada (linha 19).

A fim de distinguir entre as duas equações propostas (Equações 3.2 ou 3.3), esta dissertação referencia a abordagem explorando cidades e atrações (i.e., utilizando a Equação 3.3) como ReCWEE+. É importante notar que, embora ReCWEE+ seja explicada com base no cenário *cidades* $\times$ *atrações*, ela pode ser facilmente generalizada para outros cenários que possuem estrutura hierárquica. Por exemplo, em um sistema onde o usuário avalia respostas e se deseja recomendar categorias de respostas (cenário *categorias* $\times$ *respostas*). Um exemplo de sistema que possui esta hierarquia é o Quora [5], o qual categoriza suas respostas em tópicos.

[5] http://quora.com

Algoritmo 1 ReCWEE

```
1: function GraphGeneration(U, τ)
2:     Graph ← ∅
3:     for u1 in U do
4:         for u2 in U do
5:             if u1 ≠ u2 then
6:                 Graph[u1][u2] ← Jaccard(Cities(u1), Cities(u2))
7:             end if
8:         end for
9:     end for
10:    Prune Graph according to threshold τ
11:    return Graph
12: end function
13:
14: function Ranking(u, Gu)
15:    CandidateCities ← Cities(Gu) − Cities(u)
16:    for c in CandidateCities do
17:        c ← score(c, u)          ▷ Equação 3.2 (ReCWEE) ou Equação 3.3 (ReCWEE+)
18:    end for
19:    rank ← SortReverseOrder(CandidateCities, score)
20:    return rank
21: end function
22:
23: function ReCWEE(u, U)
24:    τ ← 0,2
25:    Graph ← GraphGeneration(U, τ)
26:    Gu ← GenerateCommunitiesKNN(u, Graph)
27:    return top-N in Ranking(u, Gu)
28: end function
```

3.4 Recomendação de Cidades Baseada em Fatores Latentes

Nesta seção é apresentada uma abordagem de recomendação de cidades baseada em fatores latentes, a qual explora informações específicas de domínio. Essa abordagem foi construída baseando-se em um método proposto recentemente chamado SSLIM, o qual será descrito na Seção 3.4.1. Neste trabalho, o SSLIM foi adaptado para ser utilizado com diferentes fontes de evidência de informações adicionais e, por conta disso, se propõe uma família de recomendadores. As soluções propostas exploram informações textuais e geográficas e serão apresentadas em detalhes nas Seções 3.4.2 e 3.4.3, respectivamente.

3.4.1 SSLIM

O SSLIM (*Sparse LInear Methods with Side Information*) Ning & Karypis [2012] é um método baseado em fatores latentes para sistemas de recomendação top-N que usa a matriz usuário-item A e uma matriz de informações adicionais (B) para encontrar uma matriz de fatores latentes (W) que pode gerar recomendações para ele.

O SSLIM é uma extensão de um método chamado SLIM (*Sparse LInear Methods*

for Top-N Recommendation), proposto em Ning & Karypis [2011]. Ambos tentam capturar a similaridade entre itens por meio da geração de uma matriz de fatores latentes utilizando uma abordagem de fatorização de matrizes. O principal benefício desses métodos é o fato de que cada coluna da matriz de fatores latentes (W) pode ser aprendida pela resolução de um problema de otimização linear e em paralelo.

No SLIM, a pontuação estimada de um usuário u_i em relação a um item ainda não pontuado d_j ($\widetilde{a}_{ij}$) pode ser estimada como uma agregação esparsa dos itens que já foram pontuados por u_i (a_i^T):

$$\widetilde{a}_{ij} = a_i^T w_j, \tag{3.4}$$

onde w_j é um vetor coluna esparso de agregação de coeficientes e que possui tamanho n. Assim, o modelo utilizado pelo SLIM para computar as pontuações para todos os usuários/itens pode ser descrito como:

$$\widetilde{A} = AW, \tag{3.5}$$

onde $\widetilde{A}$ é a matriz das pontuações estimadas, A é a matriz binária usuário-item e W é uma matriz esparsa $n \times n$ de agregação dos coeficientes. A matriz W pode ser aprendida pela resolução de um problema linear de otimização como segue:

$$\begin{aligned} \underset{W}{\text{minimizar}} \quad & ||A - AW||_F^2 + \frac{\beta}{2}||W||_F^2 + \lambda||W||_1 \\ \text{sujeito a} \quad & W \geq 0, \\ & diag(W) = 0 \end{aligned} \tag{3.6}$$

Como uma extensão do SLIM, o SSLIM usa a mesma base teórica. Este último método também aprende W utilizando informações da matriz $usuário \times item$ A, mas ele aumenta a quantidade de informação utilizada para aprender W ao adicionar a matriz de informação adicional B no problema de otimização. No SSLIM o objetivo é aprender W satisfazendo a duas equações ao mesmo tempo $\widetilde{A} \sim AW$ e $\widetilde{B} \sim BW$, como segue:

$$\begin{aligned} \underset{W}{\text{minimizar}} \quad & ||A - AW||_F^2 + \frac{\alpha}{2}||B - BW||_F^2 + \frac{\beta}{2}||W||_F^2 + \lambda||W||_1 \\ \text{sujeito a} \quad & W \geq 0, \\ & diag(W) = 0 \end{aligned} \tag{3.7}$$

Uma vez que as colunas de W são independentes, o problema de otimização de ambos os métodos (SLIM e SSLIM) pode ser decomposto em um conjunto de problemas de otimização da forma:

$$\begin{aligned} \underset{W}{\text{minimizar}} \quad & ||a_j - Aw_j||_2^2 + \frac{\beta}{2}||w_j||_2 + \lambda||w_j||_1 \\ \text{sujeito a} \quad & W \geq 0, \\ & diag(W) = 0 \end{aligned} \tag{3.8}$$

para o SLIM e

$$\begin{aligned} \underset{W}{\text{minimizar}} \quad & ||a_j - Aw_j||_2^2 + \frac{\alpha}{2}||bj - Bwj||_2^2 + \frac{\beta}{2}||w_j||_2 + \lambda||w_j||_1 \\ \text{sujeito a} \quad & W \geq 0, \\ & diag(W) = 0 \end{aligned} \tag{3.9}$$

para o SSLIM. SSLIM e SLIM focam em dados esparsos e podem ser executados em paralelo, o que permite que sejam aplicados em sistemas reais. Como o SSLIM não possui uma implementação disponível, foi implementada uma versão robusta desses métodos que foi disponibilizada em um repositório público na Web [Bidart, 2015].

3.4.2 Informações Textuais

O SSLIM foi desenvolvido para explorar informação textual adicional associada a itens para melhorar a precisão do SLIM. O método foi testado somente no domínio de recomendação de filmes, utilizando descrições providas por especialistas como informação adicional associada aos filmes (itens). Nesta dissertação, itens (i.e., cidades), não possuem informação textual diretamente associada a eles. Em vez disso, revisões escritas por usuários são disponíveis apenas para atrações pertencentes a cada cidade, capturando a opinião dos usuários e suas impressões acerca de cada atração. Esse é um cenário diferente do cenário onde o SSLIM foi proposto por conta de duas características específicas: (i) o texto gerado não se relaciona às cidades (itens) mas sim a um nível abaixo de hierarquia, que se configuram nos sub-itens (revisões); (ii) a informação textual é proveniente de comentários curtos gerados diretamente por usuários (*User Generated Content*), o que, segundo Ghosh & McAfee [2011], é um cenário mais inclinado a conteúdo pobre em qualidade. Ao conjunto de abordagens para explorar informações textuais neste cenário denominou-se TextSSLIM, ou Hierarchical SSLIM,

de modo a diferenciar um cenário do outro. Note que a diferença entre ambos consiste apenas na forma de organização da informação que entrará no método, não na parte central do método em si, que continua sendo exatamente a mesma. No contexto do TextSSLIM, torna-se necessário que se façam algumas adaptações no modo em que as informações textuais são tratadas pelo SSLIM de modo a permitir que ele seja aplicado ao cenário deste trabalho. Em suma, deseja-se encontrar um modo de se representar *cidades* a partir das *revisões* associadas a cada uma de suas *atrações*. Por exemplo: Paris possui várias atrações tais como o Museu do Louvre e a Torre Eiffel. Centenas de revisões de usuários estão disponíveis para cada uma dessas atrações em Web *sites* como por exemplo TripAdvisor e Yelp. A pergunta que se apresenta é: como se pode encontrar uma boa representação textual de Paris a partir dessas revisões?

Em outras palavras, seja H o conjunto de todas as atrações e R o conjunto de todas as revisões. Dada uma cidade c, com um conjunto de atrações H_c, cada uma com uma lista de revisões R_a, a tarefa consiste em representar c utilizando as revisões R_a. É importante que se leve em conta que em muitos sistemas, tais como o TripAdvisor, revisões são avaliadas pelos usuários com notas no intervalo entre 1 e 5. Este trabalho argumenta que somente revisões positivas (com pontuações de 4 e 5) sobre cada atração devem ser consideradas. Isso ocorre porque essas revisões serão utilizadas em um método de similaridade entre itens e o uso de aspectos negativos poderia capturar similaridades entre itens que não são similares, já que comentários negativos tendem a ser semelhantes independentemente da categoria da atração. Além disso, mesmo o número de revisões por atração sendo altamente desbalanceado (ver Figura 3.3), com algumas atrações possuindo milhares de revisões e outras possuindo poucas, foi escolhido por manter apenas um certo número k de revisões por atração, de modo a não introduzir qualquer viés em direção a atrações específicas[6]. Dadas as revisões selecionadas em cada atração, o texto delas é então segmentado em palavras, as palavras irrelevantes são removidas e é aplicado um algoritmo de *stemming* [Alvares et al., 2005]. Os termos resultantes são então utilizados para representar as cidades como um saco-de-palavras (*bag-of-words*).

Cada termo é ponderado de acordo com sua importância para descrever a cidade. Foram experimentados quatro diferentes métricas para estimar a importância dos termos: *weighted Term Frequency* (wTF) [Belém et al., 2011], *weighted Term Spread* (wTS) [Belém et al., 2011], *Stability* (STAB) [Sigurbjörnsson & van Zwol, 2008] e TF-IDF [Baeza-Yates & Ribeiro-Neto, 1999]. As primeiras três métricas foram previamente aplicadas no domínio de recomendação de tags [Canuto et al., 2013], enquanto a última

[6]Deseja-se uma representação da cidade que seja descritiva, costruída com o maior número de atrações possível, de modo a se adequar às necessidades de diferentes usuários.

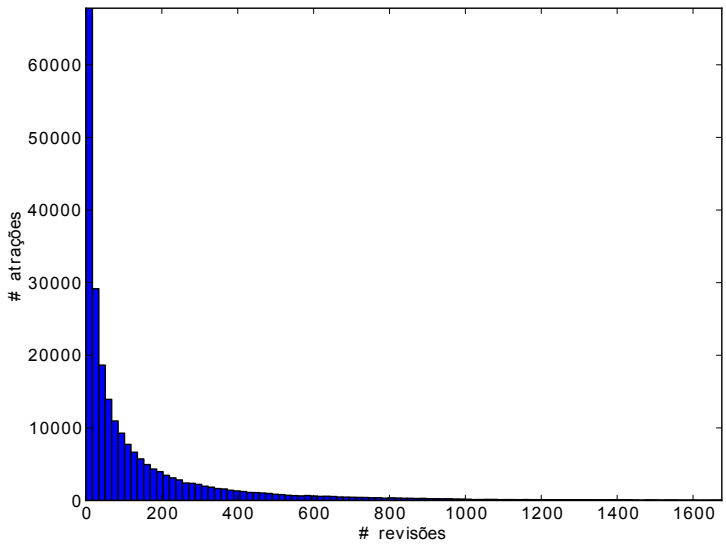


Figura 3.3: Distribuição do número de revisões por atração. Note que poucas atrações possuem muitas revisões enquanto a grande maioria das atrações possuem poucas revisões, o que denota um desbalanceamento.

é frequentemente utilizada em várias tarefas de recuperação de informação. Cada um das técnicas será descrita a seguir.

wTS. ***W**eighted **T**erm **S**pread*. No contexto de recomendação de tags, características (F) são consideradas como sendo o local onde as tags aparecem. Por exemplo, no YouTube uma tag pode ser encontrada no título ou na descrição de um determinado vídeo, onde o título é uma característica e a descrição outra. O TS (*Term Spread*) mensura a importância de uma palavra pelo número de características onde a palavra aparece. O wTS consiste em uma extensão da ideia do TS introduzindo o *Average Feature Spread* (AFS), o qual é calculado pela média do TS sobre todos os termos que aparecem em cada característica F^i. Assim, a fórmula básica do wTS denota-se por:

$$wTS(c,w) = \sum_{F^i \in F} j, \text{ onde } j = \begin{cases} AFS(F^i) & if c \in F^i \\ 0 & caso\ contrário, \end{cases} \tag{3.10}$$

No caso desta dissertação, as características foram adaptadas para serem atrações e o AFS para ser a popularidade de cada uma das atrações, popularidade esta que é dada pela média do número total de revisões que cada atração recebe. O objetivo é dar um valor maior para palavras que aparecem em diferentes atrações considerando a popularidade de cada atração, sendo que infere-se que atrações mais populares recebem

um número maior de revisões do que atrações menos populares.

wTF. ***W**eighted **T**erm **F**requency*. O wTF infere o valor de cada termo baseado em sua frequência e na importância do local em que o termo aparece. Assim, wTF é uma extensão do TF (*Term Frequency*) que usa AFS como peso para cada termo, baseado na característica na qual o termo aparece. Sua fórmula é dada por:

$$wTF(c, w) = \sum_{F^i \in F} tf(c, Fi) \times AFS(F^i) \tag{3.11}$$

Nesta dissertação, adaptou-se o TF para capturar a frequência de cada termo em uma atração específica e AFS para ser a mesma popularidade de atração utilizada no wTS. Assim, os termos recebem um valor diferente dependendo da popularidade da atração na qual são encontrados.

STAB. ***Stab**ility* é uma métrica utilizada para reduzir a importância relativa de termos que ocorrem ou muito frequentemente ou muito raramente no conjunto de treino. Assim, pode ser utilizada para representar descrições pobres de itens (i.e., cidades) [Belém et al., 2014]. O STAB de uma palavra w é calculado por:

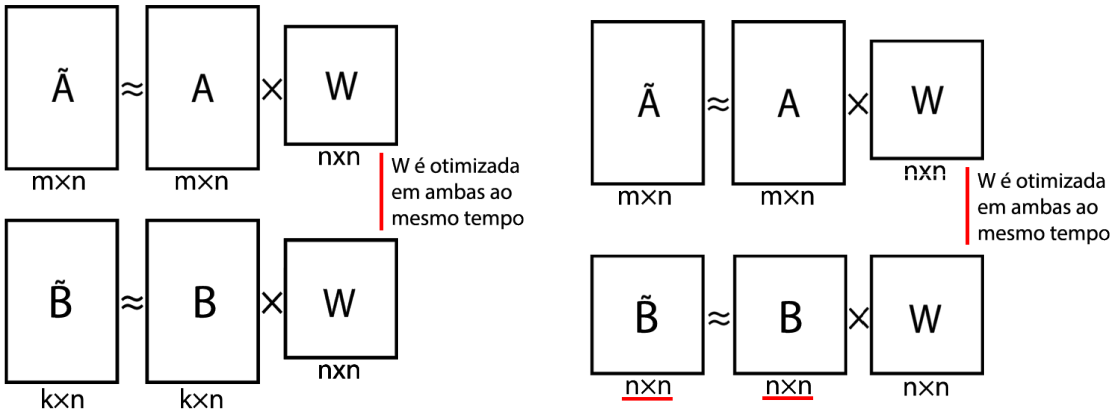
$$Stab(w, ks) = \frac{ks}{ks + |ks - log(TF(w))|}, \tag{3.12}$$

onde ks representa a "frequência ideal" de uma palavra. Nessa dissertação utilizou-se a média do TF na base como sendo o ks. Na coleção de dados, que será apresentada no Capítulo 4, o valor 16 representa a média do TF na base e foi utilizado como ks.

TF-IDF. Esta é a métrica TF-IDF padrão, considerando cidades como documentos e os termos selecionados como palavras (note que os termos são selecionados a partir do texto das revisões das atrações de cada cidade). Esta métrica captura a frequência dos termos (TF) em um mesmo documento e pondera a importância de cada termo raridade (IDF), de modo a diminuir o peso dos termos que ocorrem mais frequentemente no conjunto de textos selecionados.

3.4.3 Informações Geográficas

Informações geográficas são intrínsecas em nosso domínio, já que cada cidade possui uma localização geográfica no globo. A hipótese por trás da abordagem geográfica proposta consiste em que usuários tendem a visitar cidades que estão próximas a outras cidades que eles visitaram no passado. De fato, como apresentado na Seção 4.5.2, esta correlação foi observada na coleção de dados analisada por este trabalho. A ideia base da aplicação de informações geográficas é capturar outra dimensão de similaridade en-



(a) SSLIM (b) GeoSSLIM

Figura 3.4: Dimensões das matrizes utilizadas no SSLIM e no GeoSSLIM. Note que o SSLIM e GeoSSLIM utilizam matrizes A de mesma dimensão, no entanto, no que concerne à matriz B, SSLIM utiliza um matriz $B_{k\times n}$ enquanto GeoSSLIM utiliza uma matriz $B_{n\times n}$.

tre cidades e encontrar um método que considere a dimensão geográfica em conjunto com a dimensão comum usuário-item, de forma a gerar recomendações mais precisas. Escolheu-se então adaptar o SSLIM para agregar informação geográfica adicional ao invés de informação textual. Por conta disso, chamamos os métodos resultantes desta adaptação de GeoSSLIM. O SSLIM usa uma matriz $A_{n\times n}$ e uma matriz $B_{n\times k}$ de informações adicionais. Propõe-se então a substituição da matriz de informação adicional por uma matriz $n\times n$ de informação geográfica adicional que relaciona cidades a cidades (como apresentado pela Figura 3.4), com o objetivo de capturar as relações geográficas entre elas. Como a informação geográfica adicional e a matriz usuário-item A possuem a mesma ordem, pode-se utilizar a Equação de otimização do SSLIM 3.4.1 para resolver o problema linear do GeoSSLIM. A única diferença é que no SSLIM a matriz B possui uma dimensão $n \times k$ enquanto no GeoSSLIM a matriz B possui a dimensão $n \times n$.

Para resolver o problema de otimização enfrentado pelo GeoSSLIM foi utilizada a mesma abordagem do cSLIM [Ning & Karypis, 2012]. O cSLIM (ou **c**ollective SLIM) consiste em transformar o problema de otimização gerado pelo SSLIM no mesmo problema resolvido pelo SLIM, uma vez que existem métodos capazes de resolver o problema de otimização da Equação 3.8 utilizando *coordinate descent and soft thresholding* [Friedman et al., 2010]. A transformação feita no cSLIM consiste basicamente em transformar as matrizes A e B em uma única matriz A', da seguinte forma:

$$A' = [A, \sqrt{\alpha}B]^T \tag{3.13}$$

GeoSSLIM, SSLIM e SLIM foram implementados com base no trabalho de Levy & Jack [2013] e disponibilizados em um repositório Web com licença GPL [Bidart, 2015]. Até onde sabemos, não há implementações disponíveis do SSLIM além da que foi gerada por esta dissertação. Embora a solução algorítmica já tenha sido descrita, podem existir formas distintas para se representar a informação geográfica que compõe a matriz B. São propostas então três formas de representação distintas para as informações geográficas adicionais presentes na matriz B:

- **Bin-GeoSSLIM**: explora a característica natural do SLIM, o qual foi construído para coleções de dados com foco em *feedback* implícito (binárias). o Bin-GeoSSLIM define a relação entre duas cidades como sendo 1 se a distância entre elas fica abaixo de um limiar γ, definido em quilômetros.
- **Range-GeoSSLIM**: aplica a mesma lógica do Bin-GeoSSLIM, mas divide os dados em intervalos. Valores de distância menores do que 100 serão considerados 2. Valores entre 100 e 250 serão considerados 1 e valores maiores do que 250 serão considerados 0.
- **Dist-GeoSSLIM**: explora a distância exata entre as cidades como valor da matriz. Utiliza um limiar γ como no Bin-GeoSSLIM, de modo a manter a esparsidade da matriz B. Os valores de distância são normalizados entre 0 e 1 pela norma euclidiana e subtraídos de 1, uma vez que queremos valorizar distâncias menores entre as cidades.

Todos esses métodos são explorados para se gerar a matriz B, a qual é por sua vez utilizada como parâmetro no GeoSSLIM.

3.5 Estratégia Híbrida para Recomendação de Cidades

Os métodos propostos nesta seção partem da hipótese de que as técnicas que utilizam informações geográficas, textuais e o ReCWEE podem capturar diferentes tipos de relações entre itens. Partindo desta premissa, compreende-se que o fato das técnicas capturarem diferentes dimensões dos dados faz com que elas possam ser utilizadas em conjunto para gerar recomendações melhores e complementares para os usuários. Assim, propõe-se o uso de técnicas de RankAggregation [Dwork et al., 2001] de modo a agregar as boas características de cada uma dessas dimensões em uma só. Propõe-se utilizar diferentes métodos de agregação de modo a comparar sua eficiência no domínio

foco desta dissertação, a saber: BordaCount, Cross-Entropy e GeneticAlgorithm, tais métodos são descritos a seguir.

BordaCount. O BordaCount é um método posicional que assimila uma pontuação correspondente às posições em que cada item aparece dentro da lista ordenada de cada eleitor (i.e., cada método) e, ao final, os itens candidatos são ordenados por sua pontuação. A vantagem primária de métodos posicionais é que eles são computacionalmente baratos, podendo ser implementados em tempo linear. Eles também usufruem de propriedades como anonimidade, neutralidade e consistência, no entanto, não satisfazem o critério de Condorcet, já que não é sempre que o elemento mais votado pela maioria ganha a eleição [Dwork et al., 2001]. Basicamente o que este método faz é definir um número de itens candidatos N que é gerado a partir do tamanho da lista ordenada que será apresentada ao usuário. Dependendo da posição em que determinado item aparece em cada uma das listas ordenadas propostas por métodos distintos, ele ganha uma pontuação específica. Tipicamente essa pontuação se define como sendo $N - i$, onde i é a posição da lista ordenada em que o item aparece. Ao final, são somados os pontos recebidos pelo item em cada uma das listas em que ele aparece e este somatório é considerado como sendo a pontuação final do item avaliada pelo BordaCount. Após este processo, os itens são ordenados na lista ordenada final por ordem decrescente de sua pontuação.

Cross-Entropy. O método foi motivado por um algoritmo adaptativo para se estimar a probabilidade de eventos raros em redes estocásticas complexas, as quais envolvem minimização de variância. Diversas aplicações recentes demonstraram o poder de aplicação do método CE como uma ferramenta genérica e prática para resolver problemas NP difíceis [Pihur et al., 2007], como é o caso de se encontrar uma agregação ótima de listas. Esses métodos envolvem um procedimento iterativo onde cada iteração pode ser dividida em duas fases: (i) Geração de uma amostra de dados aleatória de acordo com um mecanismo especificado; (ii) Modificação dos parâmetros do mecanismo aleatório baseado nos dados para produzir uma "melhor" amostra na próxima iteração.

A significância do método é que define um arcabouço matemático preciso para derivar regras de aprendizado/modificação rápidas, e de algum modo "ótimas", baseadas em teoria de simulação avançada. Um maior detalhamento sobre este método pode ser encontrado em Pihur et al. [2007] e a implementação que foi utilizada neste trabalho pode ser encontrada em Pihur et al. [2009].

Genetic-Algorithm. Algoritmos genéticos são ferramentas adequadas para a resolução de problemas complexos. Sua principal vantagem consiste em sua simplicidade inerente tanto do ponto de vista do entendimento como de implementação de software [Pihur et al., 2009]. Um algoritmo genético contém tipicamente cinco etapas:

(i) inicialização; (ii) seleção; (iii) cruzamento; (iv) mutação e (v) convergência. Embora sejam uma boa solução para problemas combinatoriais, os algoritmos genéticos possuem parâmetros que, se não forem bem escolhidos, podem comprometer seu resultado. Mais detalhes sobre sua parametrização e implementação podem ser encontrados em Pihur et al. [2009].

Estes três métodos de agregação de listas ordenadas possuem formas distintas de enfrentar o mesmo problema e complexidades de solução também distintas, por conta disso foram escolhidos como métodos base para as abordagens híbridas nesta dissertação.

3.6 Sumário

Neste capítulo foi apresentada cada uma das estratégias propostas por este trabalho, as quais exploram informações adicionais hierárquicas, textuais e geográficas no domínio de turismo para recomendação de cidades. No próximo capítulo serão apresentados a metodologia utilizada para avaliação dos métodos propostos, os modelos de referência tomados como base para comparação e os resultados alcançados com cada uma das estratégias propostas.

Capítulo 4

Avaliação Experimental

Neste capítulo será apresentada a metodologia de avaliação experimental utilizada para avaliar os métodos propostos no Capítulo 3, a coleção de dados utilizada na validação deste trabalho e os resultados dos métodos propostos comparados aos modelos de referência que compõem o estado-da-arte do problema tratado nesta dissertação.

4.1 Metodologia de Avaliação

Neste estudo foi utilizada uma metodologia completamente automática de avaliação, como encontrado em outros trabalhos [Herlocker et al., 2002, 2004; Cremonesi et al., 2010; Kurashima et al., 2010; Hu et al., 2008]. Especialmente, foi adotado um modelo de validação cruzada com cinco subconjuntos. Foram utilizadas três metodologias de avaliação levemente diferentes para os métodos de vizinhança, baseados em fatores latentes e híbridos, dado que cada um possui características distintas que exigem pequenas mudanças na metodologia de avaliação para que ela se adeque ao método, embora as métricas e o modelo de validação cruzada permaneçam, em essência, os mesmos para todas as técnicas. A seguir serão apresentadas as metodologias para cada um dos métodos.

Métodos Baseados em Vizinhança. A coleção de dados foi aleatoriamente dividida em cinco subconjuntos, cada um contendo 20% das cidades e informações associadas (i.e., atração e usuários). Quatro subconjuntos foram utilizados para treino dos métodos de recomendação, ou seja, para computar as similaridades entre usuários, bem como a popularidade de cidades, e um subconjunto utilizado como teste para avaliação dos métodos. Os subconjuntos são comutados e o processo é repetido cinco vezes, cada um utilizando diferentes conjuntos para treino e teste. Perceba que, no caso particular do ReCWEE e ReCWEE+ (conforme apresentado na Seção 3.3), apenas

dados no conjunto de treino são utilizados na etapa de geração do grafo. Assim, somente cidades no treino podem ser consideradas candidatas para recomendação e o conjunto de teste é utilizado para avaliar as recomendações. Uma cidade é considerada relevante na recomendação para um usuário alvo u se ela aparece na lista de cidades visitadas por u e está no conjunto de teste, isto é, da perspectiva da estratégia de recomendação ela era até então desconhecida para u (não existia no treino), mas está presente no teste.

Métodos Baseados em Fatores Latentes. A subdivisão entre treino e teste é feita da mesma forma, no entanto há uma mudança no caso específico das matrizes complementares de informação textual e geográfica. Como seu conteúdo é associado aos itens (i.e., cidades), elas são consideradas completas contendo todo o conteúdo relativo às cidades e atrações presentes na base de dados como um todo, não segmentando treino e teste.

Métodos Híbridos A especificidade no caso dos métodos híbridos consiste no fato de eles serem aplicados apenas como etapa final de agregação dos resultados dos outros métodos propostos. Basicamente, as segmentações de treino-teste são intrínsecas para os métodos híbridos, uma vez que utilizam como base as listas recomendadas pelos outros métodos para construir sua recomendação final.

Todos os métodos são avaliados considerando a precisão média nas top-k cidades recomendadas. Seja Rec_u a lista ordenada de cidades recomendadas produzidas pelo método que está sendo avaliado para um usuário alvo u, Rec_u^k as top-k cidades nesta lista e Rel_u o conjunto de cidades relevantes para o usuário u (extraídas do conjunto de teste), a precisão nas top-k recomendações ($p@k$) é definida como:

$$p@k(Rec_u^k, Rel_u) = \frac{|Rec_u^k \cap Rel_u|}{|Rec_u^k|} \tag{4.1}$$

Todos os testes são executados cinco vezes com diferentes seleções de usuários. Os resultados são reportados utilizando um intervalo de confiança de 95%.

4.2 Coleção de Dados

A coleção de dados utilizada neste trabalho foi coletada do portal internacional de turismo *TripAdvisor* [Tri, 2015b] e corresponde a um período de um mês, de 10 de setembro de 2014 a 10 de outubro de 2014. A coleta foi direcionada para usuários brasileiros, logo foi iniciada com dois grupos como sementes: um contento as 30 cidades turísticas mais populares do Brasil e outro contendo as 15 cidades turísticas mais

populares no mundo. Essas listas de cidades mais populares foram coletadas do próprio site do TripAdvisor.

Iniciando por este conjunto de sementes, nosso *software* de coleta de conteúdo *Web* (rastreador de páginas *Web*) procede da seguinte forma: coleta primeiramente a página da cidade (veja Figura 4.1) e então segue para obter os dados de cada atração encontrada nesta cidade (veja Figura 4.2). Para cada atração, são coletadas todas as suas revisões (veja marcação 5 da Figura 4.2, onde está marcada a lista de revisões de uma atração e também a Figura 4.3 para um maior detalhamento do conteúdo de uma revisão). Depois disso, o rastreador coleta o perfil dos usuários que são autores das revisões, o qual inclui a lista de cidades previamente visitadas pelo usuário (veja Figura 4.5) e as avaliações feitas por ele (veja Figura 4.6). Finalmente o processo é repetido para cada cidade descoberta, como ilustrado pela Figura 4.4.

Figura 4.1: Página de uma cidade, as atrações estão marcadas com um traço vermelho.

A Tabela 4.1 resume as estatísticas de nossa coleção de dados, apresentando os números das principais entidades – páginas, cidades, atrações, revisões e usuários – disponíveis. Neste trabalho, foca-se em quatro dessas entidades, a saber: *cidades*, *atrações*, *revisões* e *usuários*. Para cada usuário, a coleção de dados contém um identificador, uma localização, uma lista de cidades visitadas e uma lista de atrações que ele

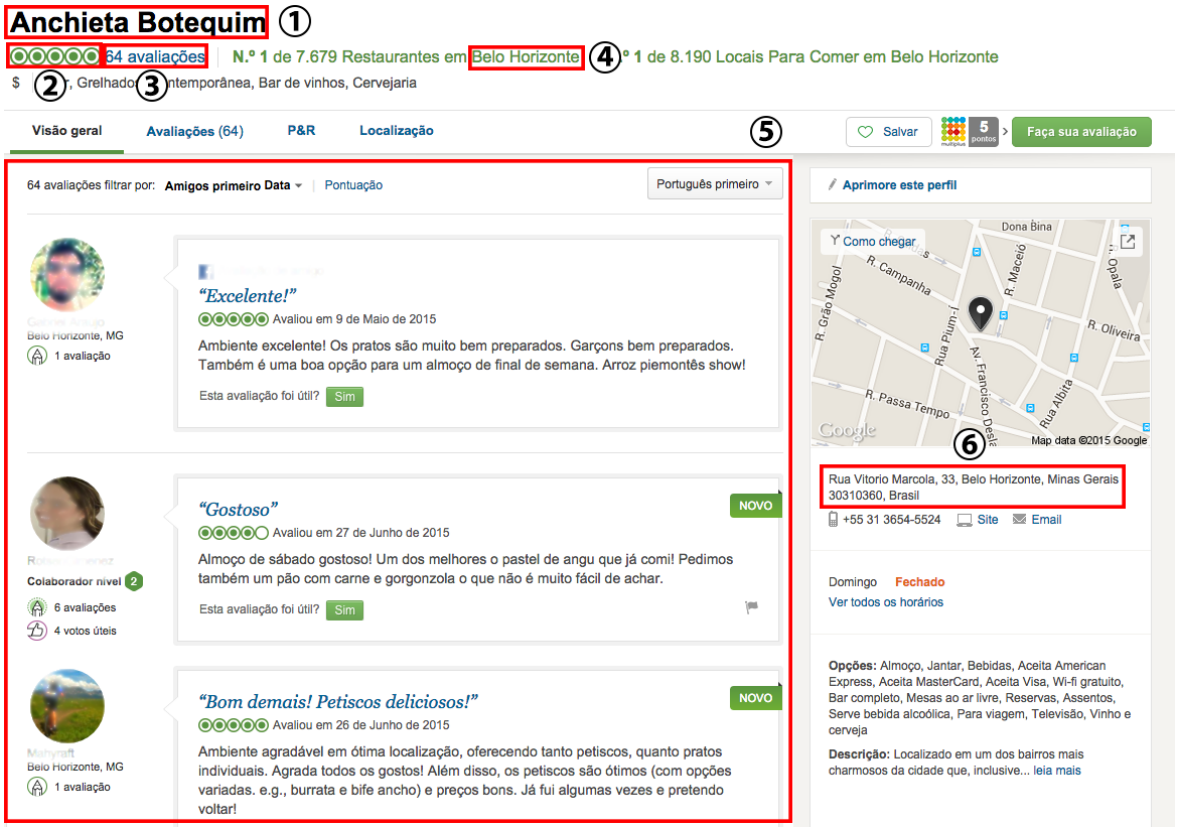


Figura 4.2: Página de uma atração. O marcação 1 indica o nome da atração, a marcação 2 sua nota, a 3 indica o número de avaliações total recebido pela atração, a 4 indica a cidade à qual a atração pertence, a 5 lista as revisões recebidas pela atração enquanto a 6 marca seu endereço.

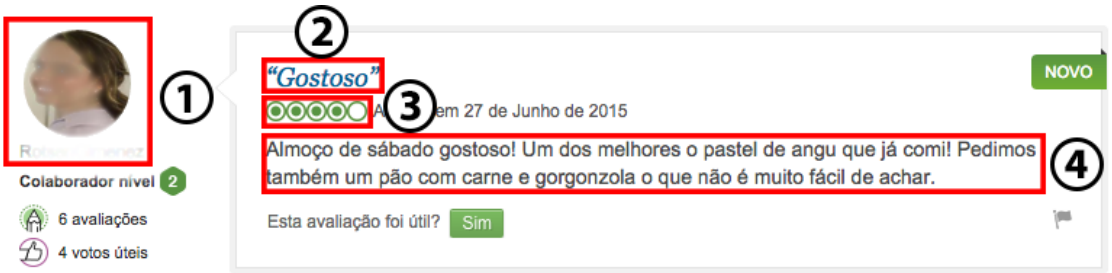


Figura 4.3: Detalhamento de uma revisão. A marcação 1 relaciona-se ao usuário que fez a revisão. A marcação 2 é o título da revisão, enquanto a 3 marca sua nota e a 4 indica o texto da revisão.

avaliou (é importante ressaltar que todos os usuários foram anonimizados por razões de privacidade). Cada atração possui um nome associado, seu endereço, sua cidade, uma média de pontuação dada pelos usuários e o número de revisões que ela recebeu. Cada cidade possui o nome e a sua localização geográfica. Os atributos disponíveis para cada

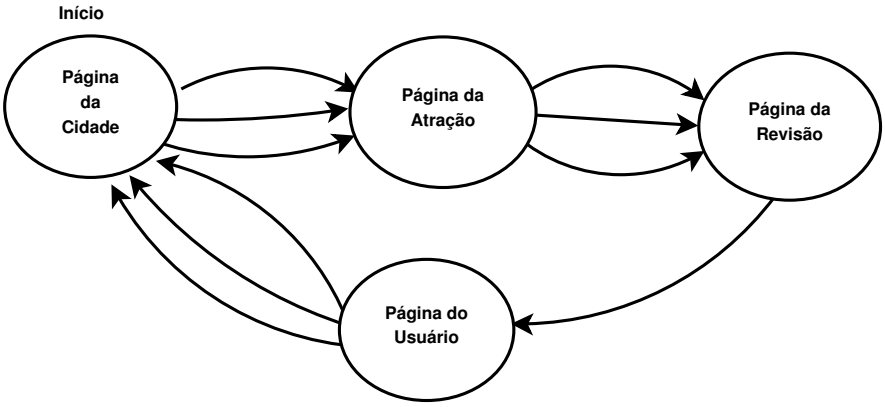


Figura 4.4: Visão geral do rastreador de páginas do TripAdvisor. Ele inicia por um conjunto de cidades (sementes) e visita todas as atrações listadas na cidade. Depois disso, coleta as revisões de cada atração. Para cada revisão apanha-se o usuário que escreveu a revisão. Finalmente, o rastreador coleta as cidades visitadas e as avaliações feitas pelo usuário, seguindo depois para as páginas dessas cidades, reiniciando o processo. Transições com uma seta significam que e página de *origem* têm somente uma entidade que leva para a página de *destino*, enquanto transições com mais de uma seta significam que a página de *origem* leva para várias entidades do *destino*.

entidade estão sumarizados na Tabela 4.2. Os dados coletados serão utilizados para avaliar todas as estratégias de recomendação propostas por esta dissertação.

Usuários	Cidades	Avaliações	Atrações	Revisões	Páginas
55,021	19,646	167,866	336,588	1,897,416	2,289,443

Tabela 4.1: Estatísticas da Coleção de Dados

Entidades	**Atributos**
Cidade	Nome, localização geográfica
Atração	Nome, cidade, nota, número de revisões, endereço
Revisão	Usuário, atração, nota, título, texto
Usuário	Identificador, localização, lista de cidades visitadas, lista de atrações avaliadas

Tabela 4.2: Atributos disponíveis por entidade

Figura 4.5: Mapa das cidades visitadas por um usuário no TripAdvisor.

Avaliações

Continuar navegando »

Avaliações recentes | Perguntas frequentes

Desde ago 2013

- Avaliações
- Fotos
- Fóruns
- Coleta de selos do TripColaboradores

Mapa de viagem

Enviar mensagem

Bloquear membro

Explore o mundo! O TripAdvisor tem dicas, avaliações e informações sobre mais de 400.000

Data de lançamento	Título	Pontuação	Avaliação útil?
21 de Junho de 2015	**João Pessoa: Pousada Jasmine Residence:** Bessa	●●●●●	0
21 de Junho de 2015	**João Pessoa: Mangai:** Melhor do Brasil	●●●●●	0
21 de Junho de 2015	**Natal: Camarões Restaurante:** Delicioso	●●●●●	1
21 de Junho de 2015	**Natal: Apê Namastê:** Muito boa. Melhor custo benefício da cidade.	●●●●●	0
21 de Junho de 2015	**Natal: Parrachos de Maracajaú:** Lindo	●●●●●	0
21 de Junho de 2015	**Natal: Dunas De Genipabu:** Super top	●●●●●	0
21 de Junho de 2015	**Barbacena: Hotel São Sebastião:** Custo benefício	●●●○○	0
21 de Junho de 2015	**Conselheiro Lafaiete: Restaurante Fogao A Lenha:** Comida caseira.	●●●●○	0
21 de Junho de 2015	**Lamim: Bar, Lanchonete e Restaurante Do Geraldo:** O melhor de Lamim	●●●●●	0
21 de Junho de 2015	**Catas Altas: Pousada Solar Da Serra:** Boa, bonita e agradável	●●●●●	0
21 de Junho de 2015	**Santa Bárbara: Bar Da Estacao:** Atendimento sofrível	●●●○○	0
21 de Junho de 2015	**Santa Bárbara: Santuário do Caraça:** Totalmente maravilhoso	●●●●●	0
21 de Junho de 2015	**Caeté: Restaurante Alpenrose:** Comida alemã.	●●●●●	0

Figura 4.6: Listagem das avaliações feitas por um usuário do TripAdvisor. As colunas da tabela mostram, respectivamente, a data em que foi feita a avaliação, seu título e a pontuação dada pelo usuário.

4.3 Modelos de Referência

Nesta seção serão apresentados os modelos de referência utilizados para avaliar o desempenho das estratégias propostas nesta dissertação. Cada modelo é então explicado em resumo, juntamente com a razão de ter sido escolhido para avaliação comparativa com os métodos propostos neste trabalho.

4.3.1 Popularidade

Popularidade consiste em uma estratégia simples e intuitiva, que sempre recomenda as cidades mais populares existentes na coleção de dados, independente do usuário alvo. Em outras palavras, esta é uma estratégia de recomendação não personalizada. A popularidade de uma cidade é estimada pelo número de usuários que já visitaram a cidade no passado. Esta abordagem, embora seja muito simples, tem se mostrado eficiente em algumas aplicações [Hu et al., 2008].

4.3.2 Item-*kNN*

Item-kNN é uma abordagem para recomendações baseada em vizinhança que é utilizada como modelo de referência em diversos trabalhos [Sarwar et al., 2001; Ning & Karypis, 2011]. Esta abordagem utiliza o conjunto de itens avaliados pelo usuário alvo e computa quão similar esses itens são em comparação com cada item d existente na base. Logo após computar as similaridades, o algoritmo seleciona os k itens mais similares dado o perfil do usuário alvo. Uma vez que os itens mais similares já foram encontrados, a posição de cada item na lista ordenada que será recomendada para o usuário é então computada levando em consideração a média ponderada das pontuações que o usuário alvo já deu para itens semelhantes. Diversas medidas de similaridade podem ser aplicadas nesses métodos, entre elas a correlação de Pearson [Boslaugh & Watters, 2008] e similaridade de cossenos [Tan et al., 2005], sendo esta última utilizada nos experimentos deste trabalho.

4.3.3 WRMF

O WRMF (**W**eighted ***R**egularized **M**atrix **F**actorization*) é considerado estado-da-arte para recomendações com *feedback* implícito, portanto é um modelo de referência forte e difícil de ser superado. Para o problema enfrentado nesta dissertação, o WRMF foi modelado com usuários e cidades (itens). O WRMF recebe as relações entre usuários e cidades (implícitas), descobre os fatores latentes e, depois disso, usa esses fatores

para recomendar cidades para os usuários. No WRMF usuários e itens são mapeados em um mesmo espaço de fatores latentes de dimensionalidade f, de modo que as interações usuário-item são modeladas como produto interno deste espaço. Cada item d é associado a um vetor $q_d \in \mathbb{R}^f$ e cada usuário u é associado a um vetor $p_u \in \mathbb{R}^f$. O produto escalar resultante $q_d^T p_u$ captura a interação entre u e d, levando à estimativa $\widetilde{r}_{ud}$ dada pela Equação 4.2.

$$\widetilde{r}_{ud} = q_d^T p_u \tag{4.2}$$

Dado que o cálculo da recomendação em si é realizado por meio de uma operação vetorial simples, o desafio real consiste em computar o mapeamento de cada vetor de fatores para itens e usuários ($q_d, p_u \in \mathbb{R}^f$). As estimativas de q_d e p_u são calculadas minimizando a seguinte equação [Koren et al., 2009]:

$$\underset{q*,p*}{min} \sum_{(u,d)\in K} (r_{ud} - q_d^T p_u)^2 + \lambda(||q_d||^2 + ||p_u||^2) \tag{4.3}$$

Ao se minimizar esta equação e encontrar os fatores latentes tem-se o modelo necessário para se realizar as recomendações utilizando WRMF.

4.3.4 SLIM

O SLIM (***S**parse **LI**near **M**ethods for Top-N Recommendations*) [Ning & Karypis, 2011] é um algoritmo que captura a similaridade entre itens por meio da geração de uma matriz de fatores latentes, utilizando uma abordagem com resolução linear em paralelo. O SLIM mostrou-se superior a diversos algoritmos estado-da-arte para recomendação Top-N [Ning & Karypis, 2011], sendo considerado forte desde então. Maiores detalhes sobre este algoritmo podem ser vistos no Capítulo 3 (Seção 3.4.1) onde o SLIM é detalhado.

4.4 Configuração Experimental

Nesta seção será apresentado, para cada algoritmo, o conjunto de parâmetros que foram utilizados para definir sua execução. Todos os algoritmos tiveram seus parâmetros calibrados por busca em grade (*grid search*) e os parâmetros escolhidos, considerando a qualidade dos resultados, são descritos a seguir e posteriomente adotados nos experimentos deste trabalho. Além disso, será apresentada a seleção de dados

que foi feita dentro da coleção de dados utilizada com o objetivo de utilizar uma base consistente nos experimentos.

4.4.1 Parametrização

Nesta subseção serão apresentados os parâmetros de cada método, as opções de valores testadas em cada um dos parâmetros, juntamente com a definição do melhor valor de parâmetro encontrado em cada caso.

- **Popularidade**: o algoritmo de popularidade não possui parâmetros.

- **ReCWEE**: os principais argumentos deste método são:

  Tamanho da Comunidade dos Usuários (k): este tamanho foi variado entre [5, 10, 15, 20, 25, 30] e o melhor resultado foi 20, mesmo valor encontrado por outros trabalhos [Herlocker et al., 2002].

  Limiar de Similaridade (τ): este limiar é naturalmente dependente da base de dados onde o ReCWEE é aplicado. Ao invés de escolher um limiar arbitrário, optou-se por analisar a distribuição dos valores de similaridade. Para este fim, gera-se um gráfico que relaciona o número de usuários removidos do grafo para vários valores de limiar. A Figura 4.7 mostra esta distribuição para a coleção de dados usada neste trabalho, a qual será apresentada no Capítulo 4. Note que a distribuição possui um joelho claro: quando o limiar τ é colocado em 0,2, somente 3,78% dos usuários são removidos, enquanto cerca de 95% das ligações são removidas. O ReCWEE procura um limiar que leva ao menor número de usuários removidos ao mesmo tempo que se remove um grande número de ligações, uma vez que estas ligações representam conexões fracas entre os usuários.

- **Item-kNN**: o algoritmo possui apenas um parâmetro k que define o número de itens vizinhos que serão levados em conta na execução da estratégia. Esse tamanho foi variado entre [5, 10, 15, 20, 25, 30] e o melhor resultado foi 20, mesmo valor encontrado por outros trabalhos [Herlocker et al., 2002].

- **WRMF**: possui dois parâmetros, λ e *número de fatores*. Estes parâmetros foram selecionados entre $\lambda = [0.015, 0.05, 0.1, 0.5, 1]$ e $número\ de\ fatores = [10, 20, 50, 80, 150, 200]$, intervalo sugerido por Koren [2008]. Foi selecionado $\lambda = 0.015$ e $número\ de\ fatores = 50$ por terem apresentado os melhores resultados.

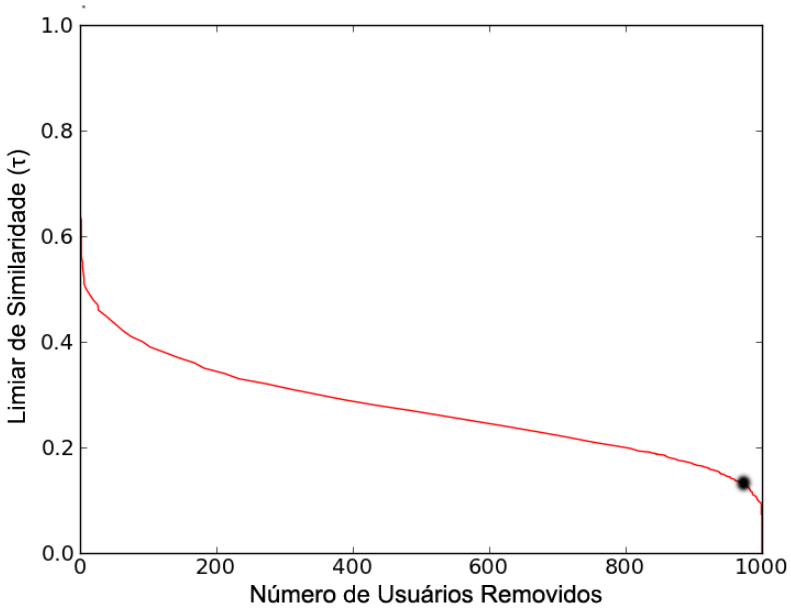


Figura 4.7: Número de usuários removidos do grafo para cada valor de limiar. Note que o joelho da curva está ao redor ce 0,2 (marcado com um círculo preto).

- **SLIM**: o método possui dois parâmetros de regularização, β e λ. Eles foram selecionados, respectivamente entre [0.001, 0.01, 0.1, 0.5, 1] e [0.0001, 0.001, 0.01, 0.1, 0.5, 1]. Selecionou-se $\beta = 0.001$ e $\lambda = 0.0001$.

- **TextSSLIM**: primeiramente é importante ressaltar que todas as métricas utilizadas neste método não são binárias, assim, foi preciso normalizar seus valores. Para esta normalização, utilizou-se a norma euclidiana [Tan et al., 2005]. Além dos parâmetros apresentados pelo SLIM, o TextSSLIM adiciona ao experimento um parâmetro α, o qual foi selecionado entre [0.01, 0.1, 0.5, 1, 2, 5]. Os valores de $\beta = 0.001$ e $\lambda = 0.0001$ aprendidos pelo SLIM foram mantidos, mas cada uma das métricas apresentou um valor distinto para α, respectivamente [2, 2, 0.1, 2].

- **GeoSSLIM**: como este método possui uma proximidade de abordagem com o TextSSLIM seus parâmetros são os mesmos. Para todas as abordagens os valores de $\beta = 0.001$ e $\lambda = 0.0001$ foram mantidos, porém, cada uma das abordagens Bin-GeoSSLIM, Dist-GeoSSLIM e Dist-GeoSSLIM foram melhores com diferentes valores de α, respectivamente [2, 1, 2]. Bin-GeoSSLIM e Dist-GeoSSLIM possuem também um quarto parâmetro – um limiar τ. Foram testados valores no intervalo [50, 100, 200, 300] e os melhores valores 100 e 200 para Bin-GeoSSLIM e Dist-GeoSSLIM respectivamente.

4.4.2 TripAdvisor 1k

Foram selecionados aleatoriamente 1.000 usuários da coleção de dados com a condição de que cada usuário tivesse visitado, no mínimo, 30 cidades. O principal objetivo desta seleção foi o de evitar o problema de *cold start*, ou seja, evitar o problema que se tem de avaliar algoritmos quando não há evidências suficientes sobre os usuários para se chegar a uma conclusão mínima acerca de suas preferências. Além disso, a seleção permitiu que a avaliação fosse feita por busca em grade, uma vez que os métodos precisaram ser executados centenas de vezes e uma base com muitos usuários poderia inviabilizar a execução de tantos experimentos, já que à medida que o número de usuários cresce, aumenta-se o tempo de execução de cada experimento.

A seleção de dados filtrou apenas os usuários, continuando a contar com o mesmo número com relação às outras entidades envolvidas, conforme pode ser visto na Tabela 4.3. Na próxima subseção, serão apresentados os resultados alcançados utilizando-se como base a TripAdvisor 1k.

Usuários	Cidades	Avaliações	Atrações	Revisões
1.000	19.646	167.866	336.588	1.897.416

Tabela 4.3: Estatísticas da coleção TripAdvisor1k

4.5 Resultados

Nesta seção são apresentados os principais resultados para cada uma das estratégias propostas e em grupos, ou seja, primeiro são apresentados os resultados para as técnicas baseadas em vizinhança e depois os resultados para aquelas que se baseiam em fatores latentes. As melhores abordagens de cada uma delas são comparadas e, posteriormente, são apresentados resultados para técnicas híbridas, as quais combinam as soluções com o objetivo de obter o melhor de ambas e assim tentar alcançar uma melhor precisão média.

4.5.1 Técnicas Baseadas em Vizinhança

Esta subseção compara as duas técnicas baseadas em vizinhança propostas por este trabalho com o modelo de Popularidade e Item-kNN, os quais são modelos de referência comumente utilizados na literatura para modelos de vizinhança [Sarwar et al., 2001]. A Tabela 4.4 apresenta a comparação entre os métodos em termos de precisão

média. Note que o ReCWEE+ apresenta-se como o melhor método. Sua distância para o ReCWEE denota o ganho existente ao se explorar as informações provenientes da segunda camada da hierarquia (i.e., atrações).

Método	@1	@3	@5	@10
Popularidade	0,121±0,010	0.140±0.007	0,135±0,004	0,124±0,003
Item-kNN	0,380±0,024	0,352±0,018	0,331±0,011	0,298±0,006
ReCWEE	0,633±0,007	0,573±0,012	0,510±0,012	0,424±0,013
ReCWEE+	**0,662**±0,012	**0,597**±0,007	**0,543**±0,007	**0,456**±0,006

Tabela 4.4: Comparação entre os métodos baseados em vizinhança com 95% de confiança. Note que o ReCWEE+ apresenta-se como o melhor método.

4.5.2 Técnicas Baseadas em Fatores Latentes

As técnicas propostas que se baseiam em fatores latentes podem ser divididas em duas classes: as que exploram informações textuais hierárquicas (TextSSLIM) e as que exploram informações geográficas (GeoSSLIM).

- **TextSSLIM**: embora as métricas propostas (wTS, wTF, STAB e TF-IDF) capturem diferentes aspectos dos atributos textuais, em termos de precisão média não foi observada uma diferença estatisticamente válida entre elas, como pode ser visto na Tabela 4.5. Como deseja-se fazer uma comparação entre essas técnicas textuais e as que exploram outras dimensões, a escolha de uma das métricas para representar a dimensão textual tornou-se necessária. Optou-se pelo wTS por ter os maiores valores médios nas posições 1 e 10 da lista ordenada.

Método	@1	@3	@5	@10
wTF-TextSSLIM	**0,744**±0,016	**0,666**±0,009	**0,602**±0,009	**0,502**±0,008
*TF*IDF-TextSSLIM*	**0,748**±0,017	**0,668**±0,008	**0,603**±0,008	**0,503**±0,008
STAB-TextSSLIM	**0,749**±0,017	**0,668**±0,008	**0,604**±0,009	**0,503**±0,008
wTS-TextSSLIM	**0,750**±0,013	**0,668**±0,008	**0,604**±0,008	**0,505**±0,007

Tabela 4.5: Comparação entre os métodos TextSSLIM. Note que os métodos não possuem resultados com diferença estatística válida entre si.

- **GeoSSLIM**: foram propostas três estratégias para explorar informações geográficas, Bin-GeoSSLIM Dist-GeoSSLIM, Range-GeoSSLIM. Os resultados para essas abordagens podem ser vistos na Tabela 4.6. Observa-se claramente que a melhor

abordagem é o Bin-GeoSSLIM, no entanto, dentro do Bin-GeoSSLIM não há um limiar (λ) entre 100, 200 e 300 que seja estatísticamente melhor do que o outro. Embora possamos notar que o Bin-GeoSSLIM(γ=300) tenha os resultados mais elevados, sua complexidade tende a ser mais elevada já que a matriz envolvida neste método é bem menos esparsa do que nas outras opções de Bin-GeoSSLIM. Assim sendo, escolheu-se o Bin-GeoSSLIM(γ=100) como melhor método pois possui um desempenho comparável ao Bin-GeoSSLIM(γ=300) a um custo menor.

Método	**@1**	**@3**	**@5**	**@10**
Dist-GeoSSLIM (γ=50)	0,753±0,015	0.672±0.009	0,609±0,013	0,506±0,007
Dist-GeoSSLIM (γ=100)	0,753±0,018	0.674±0.010	0,611±0,011	0,508±0,009
Dist-GeoSSLIM (γ=200)	0,754±0,015	0.675±0.012	0,612±0,012	0,509±0,009
Dist-GeoSSLIM (γ=300)	0,754±0,016	0,674±0,012	0,611±0,011	0,508±0,008
Range-GeoSSLIM	0,755±0,015	0,674±0,008	0,611±0,009	0,510±0,008
Bin-GeoSSLIM (γ=50)	0,773±0,013	0.690±0.009	0,629±0,010	0,525±0,008
Bin-GeoSSLIM (γ=100)	**0,779**±0,016	**0,698**±0,013	**0,638**±0,011	**0,535**±0,006
Bin-GeoSSLIM (γ=200)	**0,780**±0,011	**0,696**±0,012	**0,635**±0,011	**0,536**±0,009
Bin-GeoSSLIM (γ=300)	**0,781**±0,009	**0,699**±0,012	**0,638**±0,011	**0,538**±0,008

Tabela 4.6: Comparação entre os métodos GeoSSLIM. Note que a diferença entre os métodos Bin-GeoSSLIM a partir do limiar 100 não é estatísticamente válida. Por conta do aumento da densidade da matriz de informação adicional geográfica optou-se por selecionar o Bin-GeoSSLIM(γ=100) como melhor método GeoSSLIM.

Ao comparar-se as abordagens entre si e com os modelos de referência, é possível notar que as melhores técnicas são aquelas que exploram informações geográficas. A Figura 4.8 apresenta graficamente esta realidade.

4.5.3 Vizinhança x Fatores Latentes

Vizinhança e fatores latentes são duas abordagens distintas para o mesmo problema. A Figura 4.9 faz uma comparação de todas as melhores técnicas que exploram diferentes tipos de características dos dados do domínio. Note que as técnicas baseadas em fatores latentes são estatisticamente melhores do que as técnicas baseadas em vizinhança e que a melhor entre todas as técnicas que explora a característica geográfica do domínio (Bin-GeoSSLIM(γ=100)).

As técnicas baseadas em vizinhança têm a tendência de, naturalmente, possuírem uma menor precisão do que as que usam fatores latentes. No entanto, possuem

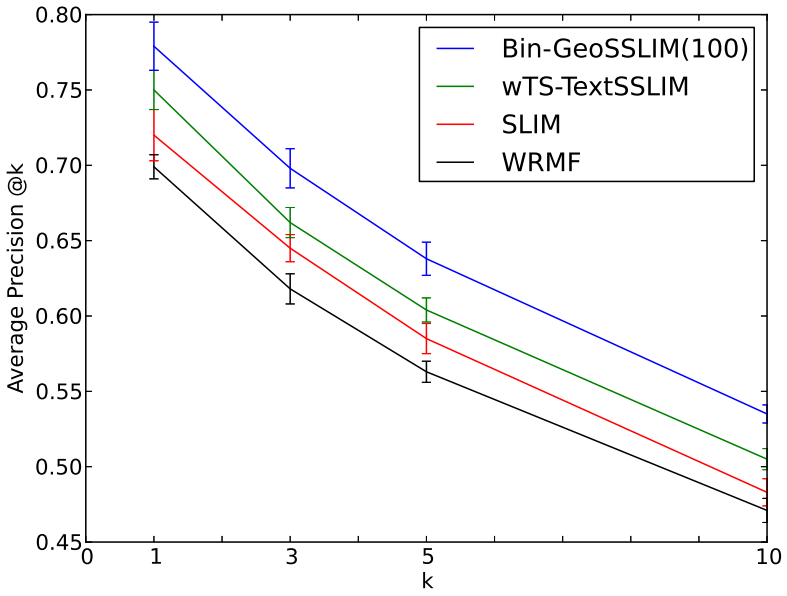


Figura 4.8: Comparação entre os métodos que exploram informações textuais e geográficas entre si e com os modelos de referência. Note que *Bin-GeoSSLIM(γ=100)* é estatísticamente melhor que os outros métodos em todas as posições da lista ordenada analisadas.

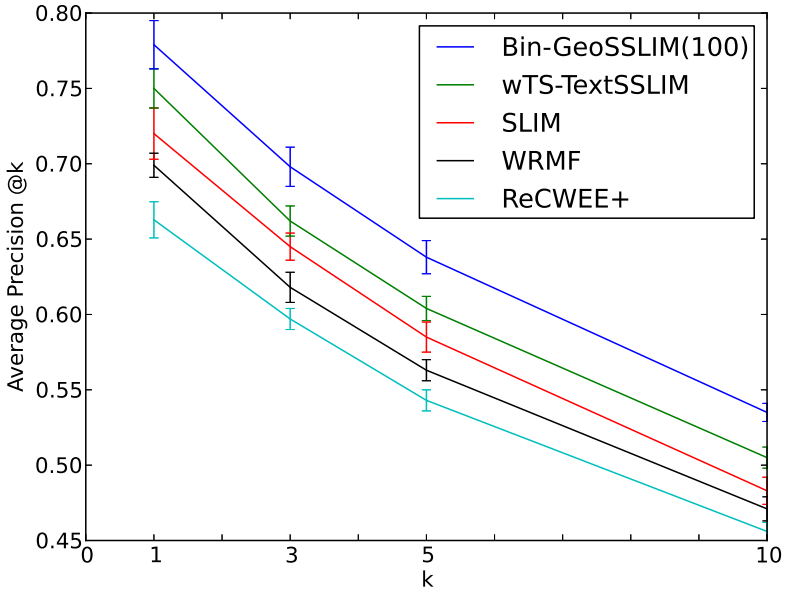


Figura 4.9

vantagens de serem facilmente interpretáveis e de executarem em um tempo menor. Na próxima seção faremos uma análise da complementariedade entre as duas técnicas.

4.5.4 Técnicas Híbridas

Foi investigado também se a combinação de fontes distintas de informação – revisões textuais e informação de localização geográfica – pode melhorar a precisão da recomendação dos destinos. Primeiramente, verificou-se o potencial dos melhores métodos que exploram as informações textuais e geográficas – wTS-TextSSLIM e Bin-GeoSSLIM, respectivamente – para recomendar diferentes itens que são considerados acertos, o que pode ser representado pela pergunta: *qual é o potencial do Bin-GeoSSLIM para recomendar itens relevantes que não são recomendados por wTS-TextSSLIM?*.

Para verificar esse potencial foram geradas, para cada usuário, as recomendações de ambos os métodos. Verificou-se a ocorrência de itens relevantes recomendados por um método, mas que não foram recomendados pelo outro. Esta noção de completude é naturalmente bem representada por um diagrama de *Venn*, como pode ser visto na Figura 4.10. A figura mostra que existe um potencial para ser explorado em uma abordagem híbrida, uma vez que os métodos capturam não somente os mesmos itens relevantes, mas também itens que são exclusivamente recomendados por cada um deles em separado. Essa capacidade exclusiva poderia ser combinada em um único método de modo a aumentar o potencial do Bin-GeoSSLIM (o melhor método proposto) utilizando informação de texto (wTS-TextSSLIM).

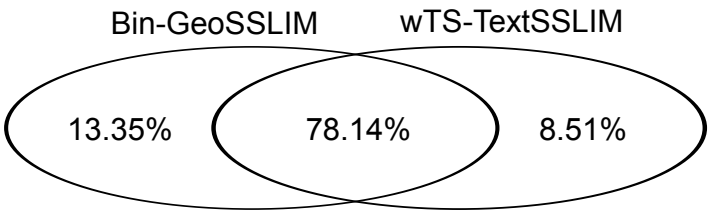


Figura 4.10: Diagrama de Venn - Potencial de agregação entre Bin-GeoSSLIM e wTS-TextSSLIM.

Investigou-se também o potencial do ReCWEE+ em relação ao Bin-GeoSSLIM, com a hipótese de que técnicas baseadas em vizinhança podem capturar outros tipos de correlação nos dados diferente das técnicas de fatores latentes. Utilizando o diagrama de *Venn*, é possível, notar na Figura 4.11, que o ReCWEE+ também é capaz de agregar novas cidades relevantes às recomendações feitas pelo Bin-GeoSSLIM, caso as listas ordenadas por cada uma das técnicas fosse combinada de forma mais eficiente. Note

que o fator de complementação do ReCWEE+ é de 23,40%, maior do que o do wTS-TextSSLIM, que é de 8,51%.

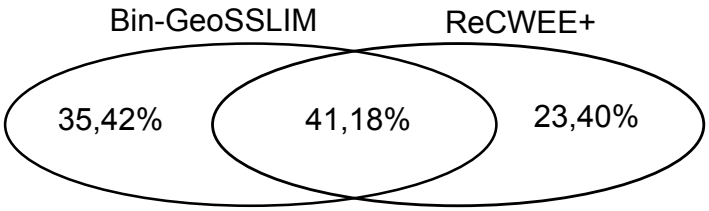


Figura 4.11: Diagrama de Venn - Potencial de agregação entre Bin-GeoSSLIM e ReCWEE+.

Para a combinação das duas técnicas que são baseadas em fatores latentes, procurou-se inicialmente a combinação das matrizes de informação adicional no método linear de otimização de matrizes. Percebeu-se então que combiná-las pelas suas matrizes seria desafiador (se for de fato possível), uma vez que as matrizes possuem diferentes dimensões. Por conta disso, a escolha foi por combinar os resultados produzidos por cada método individualmente e recomendar os top-N destinos resultantes, aplicando técnicas de *RankAggregation* [Pihur et al., 2009].

Para *RankAggregation*, foram experimentadas as técnicas BordaCount, Shulze, Genetic-Algorithm e Cross-Entropy [Pihur et al., 2009]. Entre todas as técnicas, a Genetic-Algorithm produziu os melhores resultados. Na Tabela 4.7 são apresentados os resultados de agregação entre ReCWEE+ e wTS-TextSSLIM com Bin-GeoSSLIM, usando Genetic-Algorithm. Note que, mesmo havendo um potencial comprovado pelos diagramas de *Venn*, a agregação não fornece um resultado melhor do que o Bin-GeoSSLIM em separado. Chegou-se então à conclusão que as técnicas utilizadas para agregação das listas ordenadas não foram suficientes para agregar as informações textuais e geográficas em um método apenas. No entanto, é preciso estudar profundamente as razões envolvidas com o fato, estudo este que poderá ser desenvolvido em trabalhos futuros.

Método	**@1**	**@3**	**@5**	**@10**
Bin-GeoSSLIM/ReCWEE+	0,737±0,013	0,664±0,010	0,610±0,012	0,506±0,005
Bin-GeoSSLIM/wTS-TextSSLIM	**0,762**±0,018	**0,677**±0,012	**0,619**±0,008	0,521±0,007
Bin-GeoSSLIM($\gamma = 100$)	**0,779**±0,016	**0,698**±0,013	**0,638**±0,011	**0,535**±0,006

Tabela 4.7: Comparação entre os métodos híbridos e a melhor técnica proposta neste trabalho (Bin-GeoSSLIM). Note que as técnicas híbridas não superam o Bin-GeoSSLIM sozinho, em alguns casos apenas empatam estatísticamente.

4.6 Sumário

Neste capítulo foram apresentados os resultados alcançados por este trabalho por meio de técnicas baseadas em vizinhança, baseadas em fatores latentes e, por fim, técnicas híbridas. Os resultados foram comparados ao estado-da-arte para recomendação com *feedback implícito*. Demonstrou-se que os métodos propostos são capazes de explorar informações específicas de domínio de texto e geográficas, de modo a superar os algoritmos estado-da-arte. No próximo capítulo será apresentada uma conclusão final desta dissertação, ao mesmo tempo em que serão discutidas algumas direções de trabalhos futuros.

Capítulo 5

Conclusão

Esta dissertação foca no uso de informações específicas de domínio para melhorar a precisão de recomendações top-N em sistemas de turismo. Especificamente, foi investigado o problema de sugestão de destinos para turistas, uma vez que este é o ponto inicial de um plano de viagens e é, naturalmente, uma etapa crucial para o sucesso do plano.

Foi desenvolvido um método baseado em vizinhança (ReCWEE+) e uma família de métodos baseados em fatores latentes, que se sustentam sobre uma técnica estado-da-arte (SSLIM). Foram exploradas as dimensões de informação textual e geográfica. Os métodos propostos foram avaliados utilizando uma coleção de dados real coletada do TripAdvisor, que é considerado o maior *Web* site de turismo do mundo. Os métodos baseados em fatores latentes propostos superaram dois modelos de referência fortes (WRMF e SLIM). Mais especificamente, a abordagem que utiliza informação adicional geográfica (GeoSSLIM) alcançou os melhores resultados, superando os modelos de referência em 13,32% e 9,06% (precisão@5), respectivamente. O método Bin-GeoSSLIM supera o modelo de referência baseado em popularidade em 376,1% (precisão@5), indicando que é apto a efetuar recomendações não triviais de destinos que se adequam melhor aos interesses dos usuários individualmente. Além disso, a avaliação feita mostra que o uso da localização geográfica como informação adicional tende a gerar resultados melhores do que a exploração de informação textual hierárquica (métodos TextSSLIM), considerando os dados reais do nosso cenário utilizado para validação.

Por fim, demonstrou-se que existe um grande potencial para a união das duas dimensões (textual e geográfica) em um único método, tanto pela agregação de ReCWEE+ e GeoSSLIM, quanto pela agregação do GeoSSLIM com TextSSLIM. No entanto, foi possível comprovar também que esta união não é uma tarefa trivial, uma vez que o uso de RankAggregation não foi capaz de produzir melhores resultados do que a utilização da

melhor técnica individualmente.

5.1 Contribuições Científicas

Este trabalho produziu, até o presente, algumas publicações:

1. **Para onde devo viajar? Recomendação de Cidades Baseada em Comunidades de Usuários** [Bidart et al., 2014a], trabalho que recebeu o prêmio de *Best Paper* na conferência BraSNAM;

2. ***Where Should I Go? City Recommendation Based on User Communities*** [Bidart et al., 2014b]: publicação internacional (LAWEB) de uma técnica que estende a publicação feita no BrasNAM.

Pretende-se enviar, em breve, os resultados finais obtidos e discussões do trabalho para um periódico internacional.

5.2 Trabalhos Futuros

Como trabalhos futuros, objetiva-se explorar dados publicamente disponíveis na *Web*, com intuito de ter mais informação textual sobre o contexto em estudo. Por exemplo, acredita-se que artigos da Wikipedia que descrevem cidades são uma fonte relevante de evidência textual, uma vez que tendem a ser bem escritos. Acredita-se também que a exploração de técnicas de filtragem/classificação e de sentimentos para texto é um caminho relevante a seguir para melhorar a qualidade da informação adicional de texto e combiná-la com a dimensão geográfica. Dentro do contexto desta combinação, tem-se também a possibilidade de estudo de técnicas de *RangAggregation* mais efetivas e também técnicas de escolha da melhor lista ordenada de cidades para cada usuário (como, por exemplo, o uso de *Support Vector Machines* ou mesmo *Random Forests* [Murphy, 2012]).

Além disso, serão investigadas novas dimensões de informações adicionais no domínio de turismo que podem agregar valor ao sistema de recomendação de destinos proposto. Uma informação adicional interessante e disponível para os destinos são as categorias de suas atrações. Este é um exemplo de outra dimensão específica de domínio que pode ser explorada no futuro.

Referências Bibliográficas

(2015a). Fact sheet - tripadvisor. <http://www.tripadvisor.com/PressCenter-c4-Fact_Sheet.html>. Acesso em 18 de jun. 2015.

(2015b). Tripadvisor - leia avaliações, compare os preços e reserve. Disponível em: <http://tripadvisor.com.br>. Acesso em 18 de jun. 2015.

Adomavicius, G. & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Trans. on Knowl. and Data Eng.*, 17(6):734--749. ISSN 1041-4347.

Agarwal, D. & Chen, B.-C. (2009). Regression-based latent factor models. Em *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '09, pp. 19--28, New York, NY, USA. ACM.

Agarwal, D. & Chen, B.-C. (2010). flda: Matrix factorization through latent dirichlet allocation. Em *Proceedings of the Third ACM International Conference on Web Search and Data Mining*, WSDM '10, pp. 91--100, New York, NY, USA. ACM.

Alvares, R. V.; Garcia, A. C. B. & Ferraz, I. N. (2005). Stembr: A stemming algorithm for the brazilian portuguese language. Em Bento, C.; Cardoso, A. & Dias, G., editores, *EPIA*, volume 3808 of *Lecture Notes in Computer Science*, pp. 693–701. Springer.

Ayeh, J. K.; Au, N. & Law, R. (2013). "do we believe in tripadvisor?": examining credibility perceptions and online travelers' attitude toward using user-generated content. volume 52 of *Journal of travel research : a quarterly publication of the Travel and Tourism Research Association. - Sage, ISSN 0047-2875, ZDB-ID 8643775. - Vol. 52.2013, 4, p. 437-452*, pp. 437--452. Sage. ISSN 0047-2875.

Baeza-Yates, R. A. & Ribeiro-Neto, B. (1999). *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA. ISBN 020139829X.

Balabanović, M. & Shoham, Y. (1997). Fab: Content-based, collaborative recommendation. *Commun. ACM*, 40(3):66--72. ISSN 0001-0782.

Barrington, L.; Oda, R. & Lanckriet, G. (2009). Smarter than genius? human evaluation of music recommender systems. Em *Proceedings of the 10th International Society for Music Information Retrieval Conference*, pp. 357--362, Kobe, Japan.

Batet, M.; Moreno, A.; Sánchez, D.; Isern, D. & Valls, A. (2012). Turist@: Agent-based personalised recommendation of tourist activities. *Expert Syst. Appl.*, 39(8):7319–7329.

Belém, F.; Martins, E.; Pontes, T.; Almeida, J. & Gonçalves, M. (2011). Associative tag recommendation exploiting multiple textual features. Em *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '11, pp. 1033--1042, New York, NY, USA. ACM.

Belém, F. M.; Martins, E. F.; Almeida, J. M. & Gonçalves, M. A. (2014). Personalized and object-centered tag recommendation methods for web 2.0 applications. *Inf. Process. Manage.*, 50(4):524--553.

Bidart, R. (2015). Toyslim - slim and sslim toy implementations. Disponível em: <`http://github.com/ruhan/toyslim`>. Acesso em 18 de jun. 2015.

Bidart, R.; Pereira, A.; Almeida, J. & Lacerda, A. (2014a). Para onde devo viajar: Recomendação de cidades baseada em comunidades de usuários. Em *Proceedings of the III Brazilian Workshop on Social Network Analysis and Mining (BraSNAM) in CSBC 2014*.

Bidart, R.; Pereira, A. C. M.; Almeida, J. M. & Lacerda, A. (2014b). Where should I go? city recommendation based on user communities. Em *9th Latin American Web Congress, LA-WEB 2014, Ouro Preto, Minas Gerais, Brazil, 22-24 October, 2014*, pp. 50--58.

Borrís, J.; Moreno, A. & Valls, A. (2014). Review: Intelligent tourism recommender systems: A survey. *Expert Syst. Appl.*, 41(16). ISSN 0957-4174.

Boslaugh, S. & Watters, P. A. (2008). *Statistics in a nutshell - a desktop quick reference.* O'Reilly. ISBN 978-0-596-51049-7.

Braunhofer, M.; Elahi, M.; Ricci, F. & Schievenin, T. (2013). Context-aware points of interest suggestion with dynamic weather data management. Em *Information*

and communication technologies in tourism 2014, pp. 87--100. Springer International Publishing.

Breese, J. S.; Heckerman, D. & Kadie, C. (1998). Empirical analysis of predictive algorithms for collaborative filtering. Em *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, UAI'98, pp. 43--52, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Canuto, S. D.; Belém, F. M.; Almeida, J. M. & Gonçalves, M. A. (2013). A comparative study of learning-to-rank techniques for tag recommendation. *JIDM*, 4(3):453--468.

Castillo, L.; Armengol, E.; Onaindía, E.; Sebastiá, L.; González-Boticario, J.; Rodríguez, A.; Fernández, S.; Arias, J. D. & Borrajo, D. (2008). Samap: An user-oriented adaptive system for planning tourist visits. *Expert Syst. Appl.*, 34(2):1318--1332. ISSN 0957-4174.

Ceccaroni, L.; Codina, V.; Palau, M. & Pous, M. (2009). Patac: Urban, ubiquitous, personalized services for citizens and tourists. Em *ICDS*, pp. 7–12. IEEE Computer Society.

Claypool, M.; Gokhale, A.; Miranda, T.; Murnikov, P.; Netes, D. & Sartin, M. (1999). Combining Content-Based and Collaborative Filters in an Online Newspaper.

Cremonesi, P.; Koren, Y. & Turrin, R. (2010). Performance of recommender algorithms on top-n recommendation tasks. Em *Proceedings of the 4th ACM Conference on Recommender Systems*, pp. 39--46.

Deshpande, M. & Karypis, G. (2004). Item-based top-n recommendation algorithms. *ACM Trans. Inf. Syst.*, 22(1):143--177. ISSN 1046-8188.

Dwork, C.; Kumar, R.; Naor, M. & Sivakumar, D. (2001). Rank aggregation methods for the web. Em *Proceedings of the 10th International Conference on World Wide Web*, WWW '01, pp. 613--622, New York, NY, USA. ACM.

Ekstrand, M. D.; Riedl, J. T. & Konstan, J. A. (2011). Collaborative filtering recommender systems. *Found. Trends Hum.-Comput. Interact.*, 4(2):81--173. ISSN 1551-3955.

eMarketer (2015). Travel trends for 2015: How digital will drive new opportunites for revenue and distribution. Disponível em: <http://www.emarketer.com/Webinar/7-Travel-Trends-2015/

How-Digital-Will-Drive-New-Opportunities-Revenue-Distribution/4000097>. Acesso em 18 de jun. 2015.

Fang, Y. & Si, L. (2011). Matrix co-factorization for recommendation with rich side information and implicit feedback. Em *Proceedings of the 2Nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems*, HetRec '11, pp. 65--69, New York, NY, USA. ACM.

Fortunato, S. (2010). Community detection in graphs. *Physics Reports*, 486(3-5):75 – 174. ISSN 0370-1573.

Friedman, J. H.; Hastie, T. & Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1):1--22. ISSN 1548-7660.

Garcia, A.; Torre, I. & Linaza, M. (2013a). Mobile social travel recommender system. Em Xiang, Z. & Tussyadiah, I., editores, *Information and Communication Technologies in Tourism 2014*, pp. 3–16. Springer International Publishing.

Garcia, A.; Vansteenwegen, P.; Arbelaitz, O.; Souffriau, W. & Linaza, M. T. (2013b). Integrating public transportation in personalised electronic tourist guides. *Computers & OR*, 40(3):758–774.

Garcia, I.; Sebastia, L. & Onaindia, E. (2011). On the design of individual and group recommender systems for tourism. *Expert Systems with Applications*, 38(6):7683 – 7692. ISSN 0957-4174.

García-Crespo, A.; López-Cuadrado, J. L.; Colomo-Palacios, R.; González-Carrasco, I. & Ruiz-Mezcua, B. (2011). Sem-fit: A semantic based expert system to provide recommendations in the tourism domain. *Expert Syst. Appl.*, 38(10):13310--13319. ISSN 0957-4174.

Ghosh, A. & McAfee, P. (2011). Incentivizing high-quality user-generated content. Em *Proceedings of the 20th International Conference on World Wide Web*, WWW '11, pp. 137--146, New York, NY, USA. ACM.

Gionis, A.; Lappas, T.; Pelechrinis, K. & Terzi, E. (2014). Customized tour recommendations in urban areas. Em *Proceedings of the 7th ACM International Conference on Web Search and Data Mining*, WSDM '14, pp. 313--322, New York, NY, USA. ACM.

Good, N.; Schafer, J. B.; Konstan, J. A.; Borchers, A.; Sarwar, B.; Herlocker, J. & Riedl, J. (1999). Combining collaborative filtering with personal agents for better recommendations. Em *Proceedings of the Sixteenth National Conference on Artificial Intelligence and the Eleventh Innovative Applications of Artificial Intelligence Conference Innovative Applications of Artificial Intelligence*, AAAI '99/IAAI '99, pp. 439--446, Menlo Park, CA, USA. American Association for Artificial Intelligence.

Grčar, M.; Fortuna, B.; Mladenič, D. & Grobelnik, M. (2006). knn versus svm in the collaborative filtering framework. Em *Data Science and Classification*, pp. 251--260. Springer.

Gunasekar, S. (2012). A survey on using side information in recommendation systems. Dissertação de mestrado, University of Texas at Austin.

Herlocker, J.; Konstan, J. A. & Riedl, J. (2002). An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms. *Inf. Retr.*, 5(4):287--310. ISSN 1386-4564.

Herlocker, J. L.; Konstan, J. A.; Terveen, L. G.; John & Riedl, T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, 22:5--53.

Hu, Y.; Koren, Y. & Volinsky, C. (2008). Collaborative filtering for implicit feedback datasets. Em *Proceedings of the Eighth IEEE International Conference on Data Mining*, ICDM '08, pp. 263--272.

Huang, Y. & Bian, L. (2009). A bayesian network and analytic hierarchy process based personalized recommendations for tourist attractions over the internet. *Expert Systems with Applications*, 36(1):933--943.

Hyndman, R. J. & Koehler, A. B. (2006). Another look at measures of forecast accuracy. *International Journal of Forecasting*, 22(4):679--688.

Kabbur, S.; Ning, X. & Karypis, G. (2013). Fism: Factored item similarity models for top-n recommender systems. Em *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '13, pp. 659--667, New York, NY, USA. ACM.

Kim, C. (2004). E-tourism: an innovative approach for the small and medium-sized tourism enterprises (smtes) in korea. Em *Organisation for Economic Co-operation and development (OECD). Centre for Entrepreneurship, SMEs and Local Development*, OECD '04.

Koren, Y. (2008). Factorization meets the neighborhood: a multifaceted collaborative filtering model. Em *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 426--434.

Koren, Y.; Bell, R. & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer*, 42(8):30--37. ISSN 0018-9162.

Kurashima, T.; Iwata, T.; Irie, G. & Fujimura, K. (2010). Travel route recommendation using geotags in photo sharing sites. Em *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, CIKM '10, pp. 579--588, New York, NY, USA. ACM.

Levy, M. & Jack, K. (2013). Efficient top-n recommendation by linear regression. Em *In Large Scale Recommender Systems Workshop in RecSys'13*, RecSys'2013.

Lorenzi, F.; Loh, S. & Abel, M. (2011). Personaltour: A recommender system for travel packages. Em Boissier, O.; Bradshaw, J.; Cao, L.; Fischer, K. & Hacid, M.-S., editores, *IAT*, pp. 333–336. IEEE Computer Society.

Lucas, J. P.; Luz, N.; Moreno, M. N.; Anacleto, R.; Figueiredo, A. A. & Martins, C. (2013). A hybrid recommendation approach for a tourism system. *Expert Systems with Applications*, 40(9):3532 – 3550. ISSN 0957-4174.

Miller, B. N.; Albert, I.; Lam, S. K.; Konstan, J. A. & Riedl, J. (2003). Movielens unplugged: Experiences with an occasionally connected recommender system. Em *Proceedings of the 8th International Conference on Intelligent User Interfaces*, IUI '03, pp. 263--266, New York, NY, USA. ACM.

Murphy, K. P. (2012). *Machine Learning: A Probabilistic Perspective*. The MIT Press. ISBN 0262018020, 9780262018029.

Ning, X. & Karypis, G. (2011). Slim: Sparse linear methods for top-n recommender systems. Em Cook, D. J.; Pei, J.; 0010, W. W.; Zaïane, O. R. & Wu, X., editores, *ICDM*, pp. 497–506. IEEE.

Ning, X. & Karypis, G. (2012). Sparse linear methods with side information for top-n recommendations. Em *Proceedings of the Sixth ACM Conference on Recommender Systems*, RecSys '12, pp. 155--162, New York, NY, USA. ACM.

Oard, D. W.; Kim, J. et al. (1998). Implicit feedback for recommender systems. Em *Proceedings of the AAAI workshop on recommender systems*, pp. 81--83.

Ostuni, V. C.; Di Noia, T.; Di Sciascio, E. & Mirizzi, R. (2013). Top-n recommendations from implicit feedback leveraging linked open data. Em *Proceedings of the 7th ACM Conference on Recommender Systems*, RecSys '13, pp. 85--92, New York, NY, USA. ACM.

Parra, D.; Karatzoglou, A.; Amatriain, X. & Yavuz, I. (2011). Implicit feedback recommendation via implicit-to-explicit ordinal logistic regression mapping. *Proceedings of the CARS-2011*.

Paterek, A. (2007). Improving regularized singular value decomposition for collaborative filtering. Em *Proceedings of KDD cup and workshop*, volume 2007, pp. 5--8.

Pazzani, M. J. (1999). A framework for collaborative, content-based and demographic filtering. *Artif. Intell. Rev.*, 13(5-6):393--408. ISSN 0269-2821.

Pihur, V.; Datta, S. & Datta, S. (2007). Weighted rank aggregation of cluster validation measures. *Bioinformatics*, 23(13):1607--1615. ISSN 1367-4803.

Pihur, V.; Datta, S. & Datta, S. (2009). Rankaggreg, an r package for weighted rank aggregation. *BMC Bioinformatics*, 10.

Porteous, I.; Asuncion, A. U. & Welling, M. (2010). Bayesian matrix factorization with side information and dirichlet process mixtures. Em Fox, M. & Poole, D., editores, *AAAI*. AAAI Press.

Ricci, F. (2002). Travel recommender systems. *IEEE Intelligent Systems*, pp. 55--57.

Ricci, F.; Fesenmaier, D. R.; Nader Mirzadeh, Hildegard Rumetshofer, E. S.; Venturini, A.; Wober, K. W. & Zins, A. H. (2006). Dietorecs: Travel advisory for multiple decision styles. Em *Proceedings of the CAB International Destination Recommendation Systems: Behavioural Foundations and Applications*, pp. 232--241.

Ricci, F.; Rokach, L.; Shapira, B. & Kantor, P. B., editores (2011). *Recommender Systems Handbook*. Springer. ISBN 978-0-387-85819-7.

Sarwar, B.; Karypis, G.; Konstan, J. & Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. Em *Proceedings of the 10th International Conference on World Wide Web*, WWW '01, pp. 285--295, New York, NY, USA. ACM.

Saso, K. & Biljana, P. (2012). Empirical evidence of contribution to e-tourism by application of personalized tourism recommendation system. *Annals of the Alexandru Ioan Cuza University - Economics*, 59(1):363–374.

Savir, A.; Brafman, R. & Shani, G. (2013). Recommending improved configurations for complex objects with an application in travel planning. Em *Proceedings of the 7th ACM Conference on Recommender Systems*, RecSys '13, pp. 391--394, New York, NY, USA. ACM.

Seidel, I.; Gärtner, M.; Pöttler, M.; Berger, H.; Dittenbach, M. & Merkl, D. (2010). Itchy feet: A 3d e-tourism environment. Em Sharda, N., editor, *Tourism Informatics: Visual Travel Recommender Systems, Social Communities, and User Interface Design*, pp. 209--242. IGI Global, Hershey, PA.

Shan, H. & Banerjee, A. (2010). Generalized probabilistic matrix factorizations for collaborative filtering. Em *Proceedings of the 2010 IEEE International Conference on Data Mining*, ICDM '10, pp. 1025--1030, Washington, DC, USA. IEEE Computer Society.

Sigurbjörnsson, B. & van Zwol, R. (2008). Flickr tag recommendation based on collective knowledge. Em *Proceedings of the 17th International Conference on World Wide Web*, WWW '08, pp. 327--336, New York, NY, USA. ACM.

Tan, P.-N.; Steinbach, M. & Kumar, V. (2005). *Introduction to Data Mining, (First Edition)*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA. ISBN 0321321367.

Tan, P.-N.; Steinbach, M. & Kumar, V. (2006). *Introduction to Data Mining*. Addison Wesley.

UNWTO (2014). United nations world tourism organization. Disponível em: <`http://media.unwto.org/press-release/2015-04-15/exports-international-tourism-rise-us-15-trillion-2014`>. Acesso em 18 de jun. 2015.

USTA (2014). U.s. travel answer sheet. Disponível em: <`https://www.ustravel.org/sites/default/files/page/2013/08/US_Travel_AnswerSheet.pdf`>. Acesso em 18 de jun. 2015.

van den Oord, A.; Dieleman, S. & Schrauwen, B. (2013). Deep content-based music recommendation. Em Burges, C.; Bottou, L.; Welling, M.; Ghahramani, Z. & Weinberger, K., editores, *Advances in Neural Information Processing Systems 26*, pp. 2643--2651. Curran Associates, Inc.

Vansteenwegen, P. & Souffriau, W. (2010). Trip planning functionalities: State of the art and future. *Information Technology & Tourism*, 12(4):305–315.

Wang, X. & Wang, Y. (2014). Improving content-based and hybrid music recommendation using deep learning. Em *Proceedings of the ACM International Conference on Multimedia*, MM '14, pp. 627--636, New York, NY, USA. ACM.

Xie, P.; Pei, Y.; Xie, Y. & Xing, E. P. (2015). Mining user interests from personal photos. Em *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA.*, pp. 1896--1902.

Yang, W.-S. & Hwang, S.-Y. (2013). itravel: A recommender system in mobile peer-to-peer environment. *J. Syst. Softw.*, 86(1):12--20. ISSN 0164-1212.

Ye, M.; Yin, P.; Lee, W.-C. & Lee, D.-L. (2011a). Exploiting geographical influence for collaborative point-of-interest recommendation. Em *Proceedings of the 34th international ACM SIGIR Conference on Research and development in Information Retrieval*, pp. 325--334.

Ye, M.; Yin, P.; Lee, W.-C. & Lee, D.-L. (2011b). Exploiting geographical influence for collaborative point-of-interest recommendation. Em *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '11, pp. 325--334, New York, NY, USA. ACM.

Yu, C.-C. & Chang, H.-P. (2009). Personalized location-based recommendation services for tour planning in mobile tourism applications. Em *Proceedings of the 10th International Conference on E-Commerce and Web Technologies*, EC-Web 2009, pp. 38--49, Berlin, Heidelberg. Springer-Verlag.

Yuan, Q.; Cong, G.; Ma, Z.; Sun, A. & Thalmann, N. M. (2013). Time-aware point-of-interest recommendation. Em *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, pp. 363--372. ACM.