Universidade Federal de Campina Grande
Centro de Engenharia Elétrica e Informática
Coordenação de Pós-Graduação em Ciência da Computação

Recomendações de pontos de interesse baseadas no histórico e localizações de check-ins em redes sociais baseadas em localização

Iury Dewar Cruz de Oliveira Nunes

Dissertação submetida à Coordenação do Curso de Pós-Graduação em Ciência da Computação da Universidade Federal de Campina Grande - Campus I como parte dos requisitos necessários para obtenção do grau de Mestre em Ciência da Computação.

Área de Concentração: Ciência da Computação
Linha de Pesquisa: Metodologia e Técnicas da Computação

Nome do Orientador
Leandro Balby Marinho

Campina Grande, Paraíba, Brasil

Resumo

As Redes Sociais Baseadas em Localização (RSBL) surgiram com o propósito de permitir que os usuários possam compartilhar com sua rede de amigos informações a respeito dos pontos de interesse (POIs) que eles visitaram. Neste contexto, a capacidade de recomendar novos lugares para que os usuários possam visitar é importante, pois pode, eventualmente, melhorar a experiência destes usuários ao utilizar o sistema.

O contexto geográfico certamente influencia os usuários na hora de escolher os locais a serem visitados. Sendo assim, inicialmente analisamos este contexto de forma isolada, através de recomendadores de POIs puramente baseados em informações geográficas. Além disso, propomos um novo recomendador puramente geográfico baseado em Kernels Gaussianos. Os resultados dos nossos experimentos demonstraram que o modelo proposto consegue alcançar uma maior acurácia que os recomendadores puramente geográficos presentes no estado-da-arte na maioria dos casos avaliados. Porém, esta mesma análise demonstrou que o contexto geográfico isoladamente não é capaz de gerar recomendações com alta acurácia de forma geral.

Logo, ao modelar um recomendador de POIs é necessário combinar as informações geográficas com outros contextos a fim de melhorar sua acurácia. Sendo assim, também propomos um novo recomendador de POIs que consegue capturar as preferências de usuários (de forma similar às técnicas de filtragem colaborativa) e informações geográficas em um único modelo baseado em difusão em grafos. Este recomendador visa aprender um ranking personalizado de lugares a serem recomendados para cada usuário levando em consideração os lugares visitados por outros usuários com preferências similares, as distâncias entre os lugares visitados e os lugares candidatos à recomendação, e as regiões as quais o usuário visita mais frequentemente. Os nossos experimentos mostraram que este modelo consegue ser mais eficiente que os modelos de recomendação de POIs presentes no estado-da-arte, além de conseguir alcançar uma acurácia igual ou superior às abordagens comparadas. Todos os experimentos foram realizados utilizando dados reais de umas das RSBL mais populares atualmente: o Foursquare.

Abstract

Location-Based Social Networks (LBSN) emerged with the purpose of allowing users to share, with their friends, information about points of interest (POIs) they visited. In this context, the ability to recommend new places for users to visit is important because it can eventually improve the overall user experience while using the system.

The geographical context certainly influences the locations that the users choose to visit. Therefore, initially we analyzed this context separately, through the recommenders of POIs purely based on geographical information. Furthermore, we propose a new geographic-aware recommender based on Gaussian Kernels. The results of our experiments demonstrated that the proposed model can achieve higher accuracy than the state-of-the-art recommenders solely based on geographical information, in most of the cases evaluated. However, this same analysis showed that the geographical context alone is not able to generate recommendations with high accuracy.

So to model a new recommender of POIs, it is necessary to combine geographic information with other contexts in order to achieve high accuracy. Thus, we also propose a new recommender of POIs that can capture the preferences of users (similar to collaborative filtering techniques) and geographical information in a single model based on diffusion on graphs. This recommender aims to learn a personalized ranking of places to be recommended for each user taking into consideration the places visited by other users with similar preferences, the distances between the places visited and places candidates for recommendation, and the regions which the user visits more often. Our experiments showed that this model can be more efficient than state-of-the-art recommenders of POIs, also achieving an accuracy equal to or greater than the compared approaches. All experiments were conducted using real data from one of the most popular RSBL nowadays: Foursquare.

Agradecimentos

Agradeço a Deus, ao meu orientador, aos meus familiares e aos meus amigos.

Conteúdo

1 Introdução **1**

1.1 Motivação 1

1.2 Objetivos 5

1.3 Formalização do Problema 5

1.4 Contribuições 6

1.5 Estrutura da Dissertação 7

2 Trabalhos Relacionados **8**

3 Metodologia **13**

3.1 Descrição dos Dados 13

3.2 Protocolo de Avaliação 14

4 Análise do Impacto do Contexto Geográfico **17**

4.1 Recomendação baseada em Proximidade de POIs 18

4.2 Recomendação baseada na Localização da Residência do Usuário 19

4.3 Recomendação baseada em Regressão Linear 20

4.4 Recomendação baseada em Múltiplos Centros Gaussianos 22

4.5 Recomendação baseada em Kernels Gaussianos 23

4.6 Experimentos e Resultados 25

5 Modelo de Difusão para Recomendação de POIs **28**

5.1 Passeio Aleatório 29

5.2 Modelagem dos Dados de Check-in em um Grafo 33

5.3 Difusão Baseada em Filtragem Colaborativa 35

5.4 Difusão Baseada em Distâncias 37
5.5 Difusão Baseada em Regiões 39
5.6 Modelo Unificado de Difusão 41

6 Validação do Modelo de Difusão **45**
6.1 Abordagens Comparativas 46
6.2 Avaliação da Acurácia 47
6.3 Avaliação do Desempenho 50

7 Conclusões **52**

Lista de Figuras

1.1 Check-ins de usuários 4

3.1 Distribuição Global de Check-ins no Foursquare 14
3.2 Quantidade de Check-ins por Cidade 15

4.1 Ilustração de POIs em uma cidade 19
4.2 Exemplo de Cálculo da Localização da Residência de um Usuário 20
4.3 Distribuição das Distâncias entre Check-ins (escala logaritmica) 21
4.4 Half-Normal Q-Q Plots 24
4.5 Kernels Gaussianos 25
4.6 Resultados dos Recomendadores Puramente Geográficos 27

5.1 Grafo Dirigido e Valorado 30
5.2 Grafo representando os check-ins de usuários em lugares 34
5.3 Grafo Baseado em Filtragem Colaborativa 37
5.4 Grafo Baseado em Distâncias 39
5.5 Grafo Baseado em Regiões 41
5.6 Grafo Unificado 43

6.1 Resultados da Acurácia dos Recomendadores 47
6.2 Resultados do Desempenho dos Recomendadores 51

Lista de Tabelas

3.1 Características das Bases das Cidades 15

5.1 Iterações do Passeio Aleatório 32

6.1 Precision@5 na cidade de Chicago 49

6.2 Teste T de Student Pareado - Hipótese Nula: UG >= DGM 49

6.3 Teste T de Student Pareado - Hipótese Nula: UG = DGM 50

Capítulo 1

Introdução

1.1 Motivação

Os dispositivos equipados com GPS (Global Positioning System), tais como smartphones e câmeras digitais, estão cada vez mais acessíveis aos consumidores. Isso fez com que a quantidade de dados georreferenciados, isto é, que contém metadados a respeito da posição geográfica do dado, crescesse consideravelmente nos últimos anos. Podemos encontrar fotos georreferenciadas em serviços como o Flickr [1] e microtextos georreferenciados em serviços como o Twitter [2], por exemplo.

Dentro do contexto de dados georreferenciados, pudemos ver nos últimos anos o surgimento e popularização das RSBL (Redes Sociais Baseadas em Localização) tais como o Foursquare [3] e o Facebook places[4]. O principal tipo de dado compartilhado em uma RSBL é chamado de *check-in*. Um usuário geralmente faz check-in em uma RSBL para compartilhar com seus amigos sua atual localização. Juntamente com o dado da localização, os usuários geralmente podem incorporar texto ao check-in, permitindo assim que um dado usuário compartilhe não apenas onde ele está, mas também comentários acerca do local onde ele está. Além de ajudarem os usuários a descobrirem novos lugares a serem visitados, as RSBL também auxiliam estes usuários a promoverem o encontro (físico) com seus amigos da rede, pois quando um usuário compartilha sua localização, seus amigos podem perceber

[1] https://www.flickr.com/

[2] https://twitter.com/

[3] https://foursquare.com/

[4] http://www.facebook.com/about/location

que ele se encontra perto, ou em algum local que eles gostam de ir, e optarem por irem se encontrar com ele. Estes são alguns dos fatores que podem explicar a evidente popularização das RSBL nos últimos anos. O Foursquare, por exemplo, foi lançado em 2009 e em 5 anos atingiu as marcas de mais de 50 milhões de usuários e de 6 bilhões de check-ins [5].

Os sistemas de recomendação podem ser bastante úteis para melhorar a experiência dos usuários das RSBL, sobretudo em grandes cidades, onde há uma grande variedade de POIs (Pontos de Interesse) para visitar tais como restaurantes, museus, parques, cinemas, teatros, entre tantos outros. Quando as opções de lugares a serem visitados é grande, os sistemas de recomendação podem auxiliar os usuários a encontrarem informações relevantes, através de recomendações personalizadas de locais que eles ainda não visitaram, mas que provavelmente teriam interesse de visitar. Desta forma, os usuários podem se surpreender positivamente com o sistema em torno da RSBL, o que pode levar a um aumento no grau de satisfação e de fidelidade dos usuários.

Uma das abordagens mais utilizadas para a implementação de sistemas de recomendação é a de filtragem colaborativa [KBV09], que assume que usuários que apresentaram comportamentos similares no passado tendem a apresentar comportamentos similares no futuro. Porém, um dos grandes desafios enfrentados pelos modelos de filtragem colaborativa, de uma forma geral, é a esparsidade dos dados. Este problema diz respeito ao fato de que apesar de existir uma quantidade muito grande de itens disponíveis no sistema, os usuários tendem a interagir ou avaliar apenas uma pequena parcela destes itens. Quanto mais esparsos são os dados, mais difícil é para o modelo de recomendação aprender as preferências dos usuários do sistema. Logo, a esparsidade dos dados tem um impacto direto na acurácia do recomendador.

Limitações geográficas tornam o problema de esparsidade dos dados ainda mais agravante quando estamos no domínio das RSBL, pois um usuário nem sempre poderá visitar os POIs que ele gostaria de visitar. Por exemplo, fica claro que mesmo que um dado usuário tenha bastante interesse em visitar a Torre Eiffel em Paris, França; isso não implica que ele irá fazê-lo, sobretudo se este usuário morar em outro país. Afinal, viajar demanda tempo e dinheiro. Em alguns outros domínios este problema é menos agravante pois não existem razões óbvias que impeçam um usuário de assistir o filme Titanic caso ele tenha vontade, por

[5] https://foursquare.com/about

exemplo. Logo, o histórico de visitas dos usuários das RSBL estarão geralmente limitados por fatores geográficos, o que faz com que seja mais difícil para os modelos de recomendação aprenderem o gosto pessoal de cada usuário.

Sendo assim, neste trabalho nós começamos isolando o fator geográfico dos POIs e usuários e investigamos seu impacto nas recomendações de POIs. Pois na literatura revisada não encontramos nenhum trabalho que realizasse tal investigação. Além de analisar os recomendadores de POIs puramente baseados em fatores geográficos presentes no estado-da-arte, nós propomos um novo recomendador puramente geográfico baseado em Kernels Gaussianos. Os resultados desta análise mostraram que o recomendador de POIs proposto baseado em Kernels Gaussianos alcançou uma maior acurácia que os demais recomendadores puramente geográficos na maioria das cidades avaliadas. Porém, esta análise também demonstrou que isoladamente os recomendadores puramente geográficos, apesar de importantes, não apresentam alta acurácia. Logo, as informações geográficas devem ser exploradas juntamente com outros contextos.

Esta pode ser uma das razões que levaram trabalhos anteriores a combinarem modelos de filtragem colaborativa, modelos cientes de contexto geográfico e modelos cientes de contexto social para gerar recomendações de POIs [CYKL12; YYLL11; YSC+13]. Apesar do fato de que a combinação destes modelos tende a elevar a acurácia das recomendações, esta abordagem apresenta algumas desvantagens; dentre elas podemos citar o fato de que é necessário treinar e ajustar os parâmetros de mais de um modelo para que as recomendações possam ser efetuadas. Além disso, pelo fato de cada modelo ser independente, esta abordagem de combinação de modelos não permite explorar as correlações que podem existir entre os diferentes aspectos do domínio.

Então, propomos um modelo unificado baseado em difusão em grafos que leva em consideração as similaridades entre preferências de usuários, a influência geográfica através das distâncias entre POIs (isto é, a preferência por locais fisicamente próximos) e a preferência de usuários por determinadas regiões. Para que fique mais claro a importância de cada uma destas três relações para a recomendação de POIs, considere o exemplo de descrito na figura 1.1. Nela podemos observar um conjunto de quatro usuários (u_1, u_2, u_3 e u_4). Cada um destes usuários pode fazer check-ins em 8 localizações, que estão dispostas em duas regiões distintas: a região verde (composta por l_1, l_2, l_3 e l_4) e a região azul (composta por l_5,

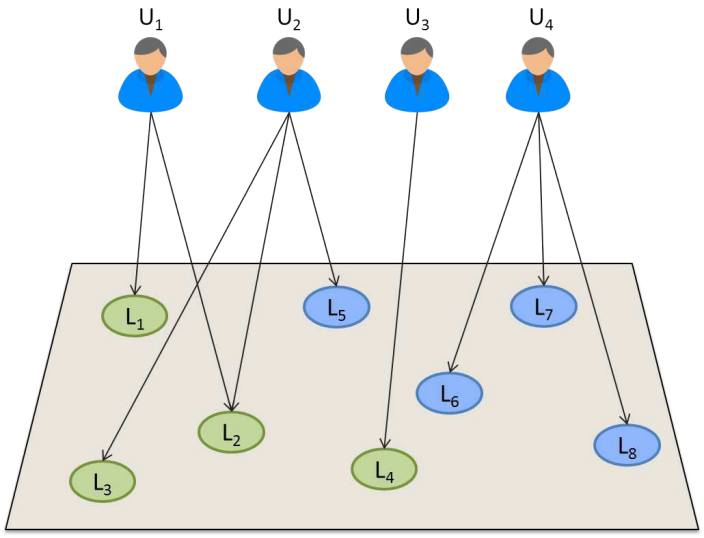


Figura 1.1: Check-ins de usuários

l_6, l_7 e l_8). As setas indicam que um usuário fez check-in naquela localização, por exemplo o usuário u_1 fez check-in nas localizações l_1 e l_2.

Daí, podemos observar que a preferência dos usuários pode ajudar a gerar recomendações para os usuários u_1 e u_2, tendo em vista que eles visitaram uma mesma localização (l_2). Se assumirmos que pessoas que visitaram localizações similares no passado tendem a visitar localizações similares no futuro, então podemos recomendar l_5 e l_3 para o usuário u_1. Similarmente, podemos recomendar l_1 para o usuário u_2. Já a preferência por locais mais próximos pode nos ajudar a gerar recomendações para o usuário u_3, que visitou apenas a localização l_4. Assumindo que usuários tendem a visitar locais próximos daqueles que ele já visitou no passado, uma potencial recomendação para u_3 seria l_6, que é a localização mais próxima de l_4. Finalmente, utilizando as preferências por regiões seremos capazes de recomendar l_5 para u_4, pois u_4 visitou todas localizações da região azul, exceto l_5. Se considerarmos que cada região é, por exemplo, um bairro de uma dada cidade, o padrão de check-ins de u_4 demonstra que ele tem bastante interesse em visitar lugares daquele bairro, justificando assim a recomendação de l_5. No modelo proposto neste trabalho, utilizamos um algoritmo de passeio aleatório para que possamos capturar, de forma transparente, o efeito de cada um destes três tipos de preferências individualmente, bem como o efeito que elas exercem umas sobre as outras.

1.2 Objetivos

O objetivo geral deste trabalho é melhorar as recomendações de POIs no contexto de RSBL. Mais especificamente, queremos definir um modelo baseado em difusão em grafos e avaliá-lo, utilizando dados reais de RSBL, sob a hipótese de que este modelo proposto apresenta uma melhor acurácia que os modelos de recomendação de POIs presentes no estado-da-arte.

1.3 Formalização do Problema

O cenário de recomendação que exploramos neste trabalho é aquele onde conhecemos com precisão a cidade na qual o usuário está localizado atualmente, independente dele morar naquela cidade ou apenas estar visitando a mesma. Neste cenário, desconsideramos a possibilidade de recomendar POIs que não estão localizados na cidade atual do usuário. A relevância deste cenário de recomendação reside no fato de que é mais viável para um usuário visitar localizações próximas de onde ele está, isto é, provavelmente será de maior interesse para um dado usuário que ele receba recomendações da cidade que ele mora (ou que está visitando) do que receber recomendações de lugares totalmente distantes de sua atual localização.

Daí, podemos definir formalmente o problema de recomendação de POIs dentro de uma cidade da seguinte forma: Seja U o conjunto de usuários, L o conjunto de lugares (POIs) e C o conjunto de cidades, então um check-in pode ser modelado através de uma tupla (u, l, c) onde $u \in U$, $l \in L$ e $c \in C$. Sendo assim, a tarefa do recomendador de POIs é definir uma função $\hat{s} : U \times L \times C \rightarrow \mathbb{R}$ que seja capaz de calcular a preferência de um dado usuário u com relação a uma dada localização l, situada numa dada cidade c. Agora, podemos calcular um ranking com as top-N localizações a serem recomendadas para um dado usuário u, em uma dada cidade c, como descrito na equação 1.1.

$$\textit{topN}(u, c) := \underset{l \in L_c \setminus L_u}{\overset{n}{\operatorname{argmax}}} \hat{s}(u, l, c) \tag{1.1}$$

onde n representa o número de POIs a serem recomendados, L_c é o conjunto de POIs localizados dentro de uma cidade c e L_u é o conjunto de lugares onde o usuário u já fez check-in.

1.4 Contribuições

As principais contribuições deste trabalho estão sumarizadas a seguir:

- Um novo recomendador puramente geográfico baseado em Kernels Gaussianos.
- Uma análise comparativa entre vários modelos de recomendação baseados em informações puramente geográficas a fim de avaliar o quão acurados são os mesmos.
- Um novo modelo de recomendações personalizadas de POIs que, a partir dos dados de check-ins de usuários de RSBL, consegue capturar, em uma única estrutura, as preferências dos usuários por locais e regiões, além de levar em consideração as distâncias entre os POIs já visitados pelo usuário e os POIs candidatos. Exploramos as influências que cada um destes três tipos de informações exercem uns sobre os outros através de um modelo de difusão em grafos: o passeio aleatório.
- A abordagem proposta foi avaliada utilizando dados reais de uma das RSBL mais populares atualmente: o Foursquare. Os experimentos realizados mostraram que os recomendadores propostos apresentam acurácia igual ou superior aos modelos de recomendação de POIs presentes no estado-da-arte, além de apresentarem um melhor custo computacional na geração das recomendações.

As duas primeiras contribuições listadas anteriormente, isto é, a definição de um novo recomendador puramente geográfico baseado em Kernels Gaussianos bem como a análise comparativa entre os modelos puramente geográficos, foram publicadas no artigo "A Gaussian Kernel Approach for Location Recommendations", que foi apresentado no simpósio KDMILE (Symposium on Knowledge Discovery, Mining and Learning) do ano de 2013. As demais contribuições geraram o artigo "A personalized geographic-based diffusion model for location recommendations in LBSN", que foi aceito para publicação no congresso LA-WEB (Latin American Web Congress) do ano de 2014. Ambos os artigos tiveram como autores o autor desta dissertação (Iury Nunes) e seu orientador (Leandro Marinho).

1.5 Estrutura da Dissertação

O restante desta dissertação está estruturada como descrito a seguir. No capítulo 2 apresentamos os trabalhos relacionados existentes na literatura e posicionamos este trabalho em relação aos mesmos a fim de justificar sua inovação e relevância. Em seguida, no capítulo 3, realizamos uma análise descritiva dos dados de check-ins a serem utilizados nos experimentos, descrevemos o protocolo de avaliação e introduzimos as métricas que serão utilizadas para medir a acurácia dos recomendadores. No capítulo 4, realizamos uma análise comparativa entre recomendadores puramente geográficos. No capítulo 5 propomos um novo recomendador de POIs, que modela os dados de check-ins em um grafo cujo os nós que representam três tipos distintos de entidades: usuários, lugares e regiões; e mostramos que aplicando os princípios de difusão em grafos através do passeio aleatório somos capazes de encontrar os lugares mais relevantes para recomendar aos usuários, levando em consideração os relacionamentos entre estas três entidades. No capítulo 6 descrevemos como validamos este modelo de difusão. Inicialmente detalhamos as técnicas de recomendação de POIs que constituem o estado-da-arte e que, portanto, foram utilizadas nos nossos experimentos. Em seguida, apresentamos e analisamos os resultados obtidos com os experimentos realizados, que mostram que, na maioria dos casos, o modelo de difusão é mais acurado e mais eficiente computacionalmente que os modelos do estado-da-arte. Por fim, o capítulo 7 finaliza a dissertação mostrando as principais conclusões alcançadas bem como apresentando idéias para potenciais trabalhos futuros.

Capítulo 2

Trabalhos Relacionados

Neste capítulo iremos descrever os trabalhos que de alguma forma se relacionam com o problema de melhorar as recomendações de POIs em RSBL. Estes trabalhos foram encontrados por meio de pesquisas nas principais conferências de Aprendizagem de Máquina e Recuperação da Informação (AAAI, SIGIR, KDD, ICWSM, SIGKDD, RecSys, CIKM, WSDM), além da bibliotecas digitais da IEEE e da ACM e sites de busca como Google Acadêmico.

Entender os padrões de mobilidade humana é um problema que vem atraindo a atenção de pesquisadores que atuam em diversas áreas, como mineração de dados espaciais e computação pervasiva. Além da recomendação de lugares, que é o foco deste trabalho, a compreensão dos padrões de mobilidade humana pode ter aplicação em diversas outras áreas, tais como, predição de condições de trânsito, planejamento urbano e criação de modelos que auxiliem o controle de doenças epidêmicas [CCLS11].

Com o surgimento e a popularização das RSBL, surgiram também bases de dados que armazenam uma série de check-ins de diversos usuários ao longo do tempo. Isso fez com que surgissem várias pesquisas que se utilizaram destes dados para analisar os padrões de mobilidade humana [CML11; CCLS11; NSMP11; SdMA$^+$14]. Apesar destes trabalhos não estarem diretamente relacionados a sistemas de recomendação, eles fornecem informações importantes que podem ser utilizadas para ajudar na concepção de novos modelos de recomendação.

Uma das conclusões destes trabalhos que foi levada em consideração para a construção do nosso modelo de recomendação de POIs é que as distâncias entre os check-ins consecuti-

vos dos usuários segue uma distribuição do tipo Lei de Potência [CCLS11; NSMP11]. Isso implica que a grande maioria dos check-ins dos usuários são realizados em locais próximos do check-in anterior. Mais especificamente, na análise feita por Noulas et al. [NSMP11] verificou-se que a distância das localizações de 80% dos check-ins consecutivos a um check-in prévio é menor que 10Km.

A distribuição de Lei de Potência também pode ser verificada quando comparamos as distâncias das localizações de qualquer check-in do usuário com a localização de sua residência. Além disso, verificou-se que tipicamente os usuário de RSBL tem pelo menos duas regiões onde a concentração de check-ins é maior: a primeira é a região ao redor de sua residência e a segunda é a região ao redor de seu local de trabalho [CML11].

O domínio das RSBL é um domínio inerentemente multirrelacional, isto é, existem várias entidades que possuem vários tipos de relações entre si. Como principais entidades, podemos citar os usuários, os POIs e a Localização dos POIs. Neste trabalho, foram exploradas dois tipos de relações encontradas neste domínio: a relação entre usuário e POIs, que é dada através dos check-ins; e a relação espacial que existe entre os POIs e suas respectivas posições geográficas. Porém, existem outras relações que são inerentes ao domínio e que podem ser utilizadas para melhorar a acurácia dos recomendadores.

A relação temporal entre o check-in e o momento em que ele foi efetuado foi explorada por Gao et al. [GTHL13] modelando a influência temporal sobre os check-ins dos usuários a partir de duas hipóteses: a primeira é que as preferências dos usuários variam ao longo do dia, por exemplo, usuários podem preferir locais mais calmos durante a manhã e locais mais barulhentos durante a noite. A segunda hipótese é que as preferências dos usuários tendem a ser mais similares em horas consecutivas do dia, ou seja, a mudança de gosto do usuário ao longo do dia se dá de forma gradual e não de forma abrupta.

Uma outra relação explorada em trabalhos recentes foi a relação entre os POIs e suas categorias [ZM12; LLAM13]. A relação de categorias dos POIs é útil para auxiliar os usuários a filtrarem conteúdo durante pesquisas de lugares a serem visitados, por exemplo, em um dado momento, um dado usuário pode estar interessado em visitar restaurantes. Parques, Shoppings, Cinemas, Estádios, Lanchonetes são outros exemplos de categorias bastante populares. Descobrir a preferência dos usuários por categorias, isto é, descobrir que um dado usuário prefere visitar shoppings enquanto um outro prefere visitar restaurantes japoneses (e

assim sucessivamente) pode ajudar a incrementar a acurácia do recomendador.

Outra relação bastante explorada em modelos de recomendação de POIs é a relação de amizade, que é dada explicitamente na Rede Social quando um usuário adiciona um outro a sua lista de amigos [CYKL12; YYLL11]. Ye et al. [YYLL11] verificou através de seus experimentos, por exemplo, que a amizade influencia check-ins de usuários em locais muito distantes de sua residência. Logo, assim como a relação temporal e a relação de categorias, um recomendador que explora a relação social de forma correta está mais propício a apresentar uma maior acurácia. Porém, estas três relações auxiliares estão fora do escopo deste trabalho. Decidimos focar apenas nas relações entre usuários e seus check-ins e entre check-ins e suas localizações, pois estas duas relações podem ser consideradas as principais relações do domínio das RSBL. Entretanto, é importante destacar que o modelo apresentado neste trabalho é robusto e flexível de forma que estas outras relações auxiliares possam ser facilmente incorporadas a ele em trabalhos futuros.

Neste sentido, os trabalhos presentes no estado-da-arte que utilizam informações destas duas relações principais, os check-ins dos usuários e a localização geográfica dos POIs, para gerar recomendação de POIs combinaram dois modelos especializados em cada uma das relações para gerar o recomendador final. Cheng et al. [CYKL12] optaram por um modelo baseado em fatoração de matrizes probabilísticas para lidar com os dados dos check-ins e por um modelo capaz de encontrar múltiplos centros com distribuição Gaussiana para encontrar as preferências geográficas dos usuários. A combinação destes modelos, porém, foi feita através de uma simples multiplicação, isto é, cada modelo é capaz de gerar um ranking atribuindo uma pontuação para cada POI, daí a pontuação de cada POI no ranking final de recomendação é a multiplicação das pontuações dos POIs nos dois ranking gerados por cada modelo. Já Ye et al. [YYLL11] utilizaram um modelo baseado nos K-Vizinhos Mais Próximos utilizando a similaridade entre usuários para determinar a vizinhança como sendo o modelo apropriado para lidar com os dados de check-in. Para lidar com os dados geográficos, foi verificado que as distâncias entre os check-ins dos usuários seguiam aproximadamente uma distribuição Lei de Potência, que pode ser aproximada por um modelo linear em escala logarítmica. Daí, foi aplicado uma regressão linear neste modelo (em escala logarítmica) a fim de encontrar os valores dos parâmetros da distribuição Lei de Potência. A combinação destes dois modelos foi feita utilizando uma combinação linear, o que implica que seria

possível dar mais peso a um modelo que a outro. Isso difere da combinação feita por Cheng et al. [CYKL12], que decidiram atribuir os mesmos pesos aos dois modelos durante a fase de combinação. A principal desvantagem de se combinar modelos através de uma combinação linear é que ela introduz novos paramêtros a serem estimados no modelo final, que é o peso de cada um dos modelos. Tendo em vista que geralmente cada modelo também tem seus parâmetros para serem ajustados, encontrar os valores ideais de todos os parâmetros do modelo final pode ser uma tarefa dispendiosa.

Diferentemente dos trabalhos mencionados anteriormente, o modelo de recomendação que apresentamos neste trabalho é unificado. Isso implica que tanto as preferências dos usuários (extraídas a partir dos check-ins) quando a influência geográfica estarão combinadas em um único modelo. E as correlações existentes entre estas duas relações distintas podem ser capturadas de forma transparente. Além disso, dado que o modelo é único, podemos focar em estimar apenas os parâmetros do modelo, sem se preocupar em depois ter que estimar pesos para combinação com outros modelos.

É importante mencionar que recentemente têm surgido alguns trabalhos que utilizaram os conceitos de modelos de tópicos probabilísticos para criar novos recomendadores de POIs [HE13; KIH+13; YSC+13]. Porém, os experimentos utilizados para a validação dos modelos propostos por [HE13] e [KIH+13] utilizaram dados de serviços como Twitter, Yelp[1] e Flickr. Apesar de todos estes serviços disponibilizarem para o usuário a possibilidade de serem postados dados georreferenciados, a qualidade dos dados para fins de recomendação de POIs pode ser considerada inferior. Por exemplo, o fato de um dado usuário ter postado uma foto no Flickr de uma dado lugar, digamos um restaurante, não quer dizer necessariamente que ele quer compartilhar com seus amigos que ele esteve naquele lugar (talvez ele tenha postado a foto por causa das pessoas que aparecem na foto). Já o trabalho desenvolvido por Yin et al. [YSC+13], utilizou dados provindos tanto de RSBL quanto de RSBE (Redes Sociais Baseadas em Eventos). Dos vários cenários explorados nesta pesquisa, existe um que mais se assemelha com o problema que estamos abordando: aquele onde usamos dados de RSBL para recomendar POIs na cidade-natal do usuário. Neste caso específico, a diferença do método de Yin et al. [YSC+13] foi bem pequena quando comparado com os resultados de Ye et al. [YYLL11]. Por estas razões, neste trabalho, para fins de validação do nosso

[1] http://www.yelp.com/

modelo, iremos comparar nosso recomendador com as abordagens definidas em [CYKL12] e [YYLL11]. Mais detalhes sobre cada uma destas abordagens podem ser encontrados na seção 6.1.

Os conceitos de difusão em grafos utilizando Passeio Aleatório já foram aplicados com sucesso para diferentes domínios. O modelo de Jaschke et al. [JMH+08] gera recomendações de Tags em serviços de bookmarking, já o modelo de Backstrom et al. [BL11] foi aplicado para a recomendação de novos amigos em uma rede social, também temos o modelo de Konstas et al. [KSJ09] que foi utilizado em recomendações musicais. Essa abordagem, de Passeio Aleatório, aparece como uma escolha bastante adequada para a recomendação de POIs pelo fato de que ela consegue facilmente explorar as diferentes relações que existem entre as entidades do domínio das RSBL. Portanto, neste trabalho iremos mostrar como podemos modelar um grafo, isto é, definir os nós, as arestas (bem como seus pesos) de forma que, ao executar o Passeio Aleatório, o modelo gerado seja capaz de capturar ao mesmo tempo, a influência definida a partir da filtragem colaborativa dos check-ins, a influência geográfica das distâncias entre os POIs e a influência geográfica das regiões onde os POIs estão situados exercidas sobre os check-ins dos usuários.

Capítulo 3

Metodologia

Neste capítulo iremos detalhar os dados que foram utilizados durante a realização dos nossos experimentos, na seção 3.1. Em seguida na seção 3.2, descrevemos como particionamos estes dados em conjuntos de treino e teste para validar os recomendadores, e também apresentamos as métricas que foram utilizadas para medir a acurácia dos recomendadores.

3.1 Descrição dos Dados

Nossos experimentos foram conduzidos utilizando dados reais de uma das mais populares RSBL atualmente: O Foursquare. Os dados do Foursquare foram coletados pelos autores de [CCLS11] no período entre setembro de 2010 e (inclusive) janeiro de 2011, e esta base está disponível (sob requisição) através do seguinte link: http://infolab.tamu.edu/data/.

Seguindo um protocolo comum encontrado na literatura para a avaliação de sistemas de recomendação, nós decidimos concentrar os experimentos em uma parte mais densa dos dados. Isso implica que desconsideramos os dados de usuários que efetuaram check-ins em menos de 10 localizações distintas, bem como desconsideramos os dados de localizações que foram visitadas por menos de 10 usuários distintos.

Após esta filtragem, as posições dos check-ins restantes no mapa mundi estão apresentadas na figura 3.1. Como podemos observar, a grande maioria dos check-ins se concentra na Europa e nos Estados Unidos. Isso pode ser explicado em parte por fatores econômicos, pois o alto poder aquisitivo dos habitantes destas regiões faz com que mais pessoas consigam ter acesso à internet móvel e a dispositivos equipados com GPS.

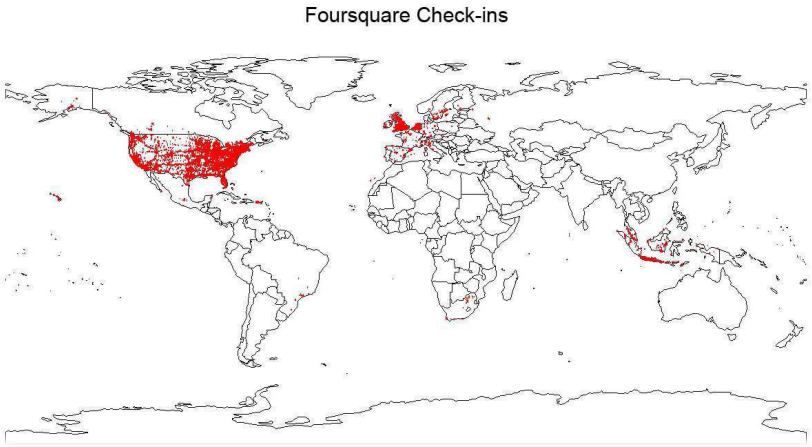


Figura 3.1: Distribuição Global de Check-ins no Foursquare

Como descrito na seção 1.3, o cenário de recomendação que exploramos neste trabalho é aquele onde conhecemos a cidade atual dos usuários. Logo, para validar os recomendadores neste contexto, é necessário escolher algumas cidades para a avaliação. Ao analisar a distribuição da quantidade de check-ins nas cidades com maior número de check-ins após a filtragem dos dados, que está exibida na figura 3.2, pudemos perceber uma grande variação na quantidade de check-ins nas 5 primeiras cidades e uma estabilização na quantidade de check-ins a partir da sexta cidade (Filadélfia). Por esta razão, para os nossos experimentos, iremos utilizar as 5 cidades com maior número de check-ins na base do foursquare após a filtragem dos dados, que são: Nova York, Los Angeles, Chicago, San Francisco e Londres. Como esperado, todas as cidades estão localizadas nos Estados Unidos ou na Europa.

3.2 Protocolo de Avaliação

Para validar os recomendadores, nós particionamos as bases de dados em dois conjuntos distintos: um conjunto para treino e um para teste. Os recomendadores devem ser capazes de estimar os itens a serem recomendados utilizando apenas os dados de treino. Caso estas recomendações incluam os dados de teste, então podemos assumir que esta foi uma boa recomendação. A proporção do particionamento foi de 90% dos check-ins dos usuários para o conjunto de treino e os 10% restantes foram usados para teste. Sendo assim, os usuários que efetuaram menos de 10 check-ins não foram considerados usuários de teste, porém todos os usuários foram considerados para o treinamento. Para cada cidade, o processo de parti-

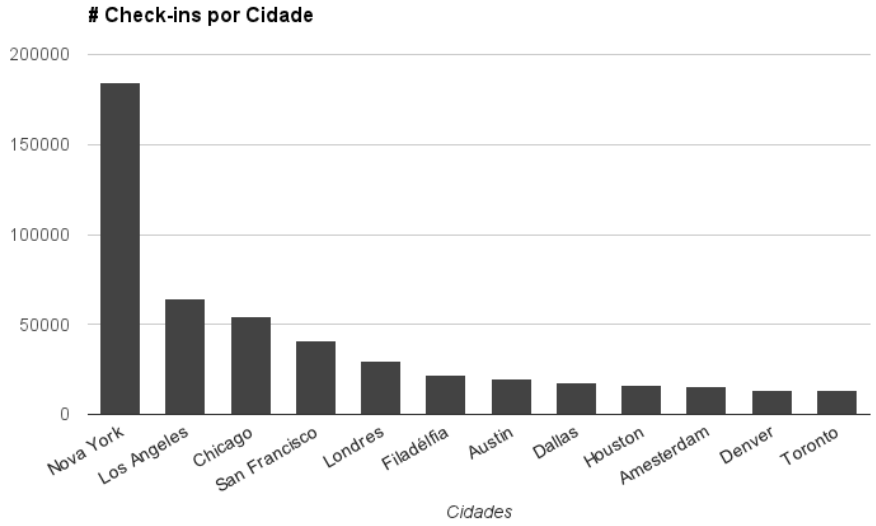


Figura 3.2: Quantidade de Check-ins por Cidade

City	# Check-ins	# Usuários	# POIs	# Usuários de Teste	Esparsidade
Nova York	184.760	12.005	3.073	4.204	$99,49\%$
Los Angeles	64.494	6.317	1.274	1.568	$99,19\%$
Chicago	54.600	5.268	1.090	1.161	$99,04\%$
San Francisco	41.148	4.877	918	939	$99,08\%$
Londres	29.992	2.861	474	794	$97,78\%$

Tabela 3.1: Características das Bases das Cidades

cionamento de 90% para treino e 10% para teste foi repetido 10 vezes com a finalidade de diminuir as chances de tirarmos conclusões de dados enviesados. A tabela 3.1 apresenta as características das bases de dados de cada uma das cidades selecionadas; onde a esparsidade dos dados é calculada conforme descrito na equação 3.1.

$$\text{Esparsidade} = 1 - \frac{\text{\# Check-ins}}{\text{\# Usuários} * \text{\# POIs}} \tag{3.1}$$

As métricas que utilizamos para medir a acurácia dos recomendadores foram o precision@5 e o recall@5. Estas métricas irão verificar a qualidade das 5 primeiras recomendações da lista. O valor 5 pode parecer pequeno, mas na prática os usuários tendem a dar uma importância muito maior para os itens recomendados no início da lista

de recomendação. Além disso, os usuários tendem a não processar grandes listas de recomendações, pois quanto maior a lista, maior o esforço cognitivo para escolher os items de interesse. Por estes motivos, 5 e 10 também são valores de referência usados na literatura [CYKL12; YYLL11; GTHL13]. Sendo assim, seja T_u o conjunto de localizações na base de testes para um dado usuário $u \in U$ e R_u as 5 primeiras recomendações da lista gerada por um dado recomendador; então podemos definir precision@5 e recall@5 para um dado usuário $u \in U$ como descrito a seguir:

$$\text{precision@5}(u) = \frac{|T_u \cap R_u|}{|R_u|}, \quad \text{recall@5}(u) = \frac{|T_u \cap R_u|}{|T_u|}$$

Capítulo 4

Análise do Impacto do Contexto Geográfico

Neste capítulo iremos realizar uma análise de comparação da acurácia de recomendadores puramente geográficos. Isto é, para cada usuário alvo (o usuário que está requisitando a recomendação) as únicas informações acessíveis aos recomendadores são as localizações dos POIs já visitados por este usuário no passado, bem como as localizações dos POIs que ele ainda não visitou. O que nos motivou a fazer esta análise foi o fato de que alguns dos recomendadores de POIs mais sofisticados presentes no estado-da-arte são, na verdade, uma combinação de modelos baseados em filtragem colaborativa, modelos cientes de contexto geográfico e modelos cientes de contexto social [CYKL12; YYLL11]. Um contexto é qualquer informação que possa ser utilizada para caracterizar a situação de uma entidade [DA00]. Neste caso, o contexto geográfico é utilizado para caracterizar a relação entre um dado usuário e um ponto de interesse. Já o contexto social é utilizado para caracterizar as relações de amizades entre os usuários.

As validações dos recomendadores de POIs presentes no estado-da-arte foram efetuadas utilizando apenas os recomendadores finais, isto é, a combinação de vários modelos. Além disso, não encontramos na literatura revisada nenhum trabalho que investigasse o impacto fator do componente geográfico na recomendação. Desta forma, não podíamos afirmar que a melhora na acurácia de um dado recomendador se deu porque o modelo ciente de contexto geográfico era mais eficiente ou se isso se deu por causa do modelo de filtragem colaborativa, por exemplo. Sendo assim, surgiu a necessidade de comparar apenas os componentes

geográficos dos recomendadores a fim de descobrir qual dos recomendadores geográficos presentes no estado-da-arte é mais acurado isoladamente.

Além disso, compararemos a acurácia destes recomendadores com um baseline bastante simples que recomenda os locais de acordo com a quantidade de usuários distintos que visitaram os mesmo. Este modelo de recomendação geralmente é referenciado pela expressão de recomendador "mais popular". A comparação com este baseline será importante para avaliar qual a real importância do fator geográfico em recomendadores híbridos que exploram este fator nas recomendações. Os experimentos foram realizados utilizando a metodologia descrita no capítulo anterior e seus resultados mostraram que, isoladamente, os recomendadores puramente geográficos não são capazes de alcançar alta acurácia.

O restante deste capítulo está organizado da seguinte forma: nas seções 4.1 e 4.2 apresentaremos dois recomendadores puramente geográficos baseados em heurísticas simples, tais como, a proximidade dos POIs entre si e a proximidade dos POIs em relação às residências dos usuários; nas seções 4.3 e 4.4 apresentaremos dois recomendadores de POIs, somente baseados em localiazação, presentes no estado-da-arte; na seção 4.5 propomos um novo recomendador puramente geográfico que, apesar de simples, consegue apresentar a melhor acurácia entre os recomendadores puramente geográficos na maioria das cidades avaliadas; finalmente, na seção 4.6 apresentamos os resultados e análises destes experimentos.

4.1 Recomendação baseada em Proximidade de POIs

Assumindo que os usuários estão mais propícios a visitarem lugares geograficamente próximos aos lugares que eles já visitaram no passado, então talvez o modelo de recomendação mais simples e intuitivo seja o de recomendar POIs baseado apenas nas distâncias entre eles. Isto é, recomendar os POIs mais próximos dos POIs já visitados por um dado usuário alvo. Assim, seja $u \in U$ o usuário alvo ao qual desejamos gerar recomendações e $l \in L$ um dado local candidato a recomendação, que está situado numa dada cidade $c \in C$, então podemos computar o grau de preferência de u por l, levando em consideração apenas as distâncias dos POIs, da forma descrita na equação 4.1.

$$\hat{s}(u, l, c) = -dist(closest(l, L_u), l) \tag{4.1}$$

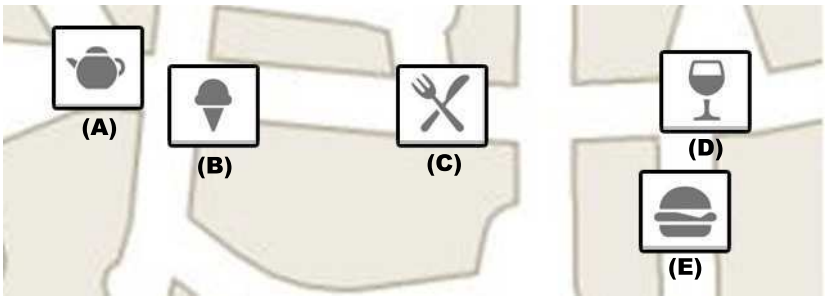


Figura 4.1: Ilustração de POIs em uma cidade

onde a função $dist$ calcula a distância (em kilômetros) entre dois lugares, e a função $closest(l, L_u)$ retorna o lugar mais próximo entre l e os POIs visitados pelo usuário u, isto é L_u. É importante destacar que multiplicamos o resultado da função $dist$ por -1 pelo fato de que quanto maior for a distância, menor a chance do usuário u visitar o lugar l.

Para que fique mais claro o processo de recomendação de um modelo baseado em proximidade de POIs, considere uma cidade que contém 5 POIs tais como descritos na figura 4.1. Agora suponha que um dado usuário visitou os POIs b e d. Sendo assim, o recomendador baseado em proximidade de POIs irá recomendar para este usuário os POIs a e e, pois eles estão mais próximos dos POIs b ou d do que o POI c. Porém se analisarmos as localizações de b e d conjuntamente, perceberemos que o POI c está localizado em uma área que é relativamente próxima dos POIs b e d, portanto recomendar c não implicaria necessariamente em uma má recomendação. Então, uma limitação do recomendador baseado em proximidade de POIs é que ele analisa cada POI visitado individualmente, desconsiderando assim qualquer interação que possa existir entre estes POIs.

4.2 Recomendação baseada na Localização da Residência do Usuário

Trabalhos recentes demonstraram que usuários tendem a visitar lugares que estão localizados próximos às suas residências [CML11]. Sendo assim, um outro recomendador puramente geográfico bastante simples que poderíamos implementar seria aquele que recomenda os lugares mais próximos da residência de cada usuário. Então o grau de preferência de um usário alvo u com relação a um POI candidato a recomendação l pode ser calculado como

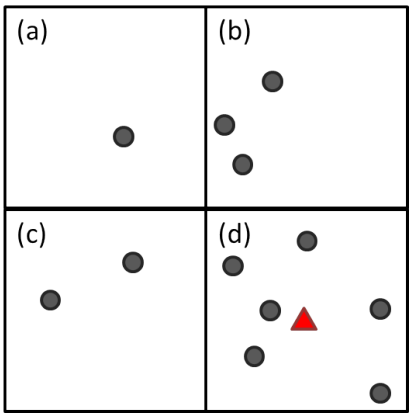


Figura 4.2: Exemplo de Cálculo da Localização da Residência de um Usuário

descrito na equação 4.2.

$$\hat{s}(u, l, c) = -dist(home(u), l) \tag{4.2}$$

onde a função $home(u)$ retorna a localização da residência dos usuários, em termos de coordenadas de latitude e longitude. Apesar dos dados referentes à residência de cada usuário não estarem explícitos em nossas bases, nós podemos inferi-los utilizando a abordagem descrita em [CML11]. Segundo esta abordagem, ao particionar o mundo em células de 25km por 25km, podemos inferir a localização de residência do usuário como sendo o centroide dos check-ins da célula na qual o usuário efetuou o maior número de check-ins.

Para que fique mais claro como as localizações das residências dos usuários são computadas, considere um dado usuário que efetuou check-ins em quatro células distintas, tal como descrito na figura 4.2. Os check-ins deste usuário estão representados pelos círculos pretos. Sendo assim, podemos perceber que a célula na qual o usuário efetuou o maior número de check-ins foi a célula d. Logo, a localização hipotética da residência deste usuário é o centroide de seus check-ins realizados na célula d, que está representado na figura 4.2 pelo triângulo vermelho.

4.3 Recomendação baseada em Regressão Linear

Em análises feitas em trabalhos anteriores, verificou-se que a distribuição das distâncias entre check-ins dos usuários é do tipo lei de potência [YYLL11], isto é, a maioria dos check-ins

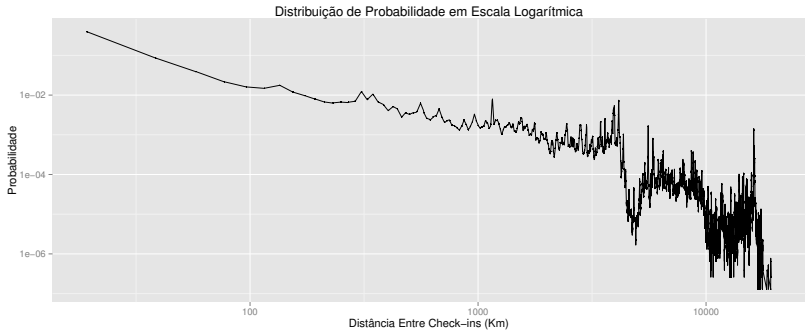


Figura 4.3: Distribuição das Distâncias entre Check-ins (escala logaritmica)

estão separados por uma distância pequena ao passo que poucos check-ins estão separados por uma distância grande. Ao realizar análises similares para os nossos dados, conseguimos chegar a conclusões semelhantes. No gráfico exibido na figura 4.3, podemos observar que a quantidade de check-ins em uma dada distância diminui de forma linear a medida que o valor da distância aumenta. Visto que a escala do gráfico está em escala logarítmica, então podemos inferir que este comportamento se assemelha a uma distribuição de lei de potência. Porém, é importante destacar que para grandes distâncias, podemos perceber uma maior irregularidade na linha do gráfico. Isso pode ser explicado pelo fato de que eventualmente os usuários viajam e passam a visitar locais muito distantes daqueles que eles visitam habitualmente [YYLL11].

Sabendo que a distribuição das distâncias entre check-ins segue uma distribuição de lei de potência, então podemos encontrar os parâmetros desta distribuição aplicando uma transformação logarítmica nos dados, conforme descrito na equação 4.3. Após esta transformação ser aplicada nos dados, obtemos uma distribuição linear com parâmetros w_0 e w_1. Então, ao aplicar a técnica de regressão linear nesta nova distribuição, encontramos facilmente os valores dos parâmetros w_0 e w_1, e através destes valores podemos encontrar os valores dos parâmetros a e b da distribuição de lei de potência original.

$$
\begin{aligned}
&P(dist(l,l')) = a * dist(l,l')^b \\
&\log P(dist(l,l')) = w_0 + w_1 * \log dist(l,l') \\
&\textit{onde } a = 2^{w_0} \textit{ e } b = w_1
\end{aligned}
\tag{4.3}
$$

Uma vez que computamos os parâmetros da distribuição de lei de potência, podemos

utilizar as distâncias entre os POIs visitados por um dado usuário para calcular o grau de preferência de um dado usuário u por uma localização l conforme descrito na equação 4.4.

$$\begin{aligned}\hat{s}(u,l,c) &= P(l|L_u) \\ &= \frac{P(l \cap L_u)}{P(L_u)} \\ &= \frac{P(L_u) * \prod_{l' \in L_u} P(dist(l,l'))}{P(L_u)} \\ &= \prod_{l' \in L_u} P(dist(l,l'))\end{aligned} \tag{4.4}$$

Observe que este recomendador baseado em regressão linear utiliza as localizações de todos os POIs que o usuário já fez check-in para calcular seu grau de preferência. Esta abordagem difere dos recomendadores descritos nas duas seções anteriores, pois o recomendador descrito na seção 4.1, calcula o grau de preferência baseado apenas em um POI (que é justamente o mais próximo), enquanto o recomendador descrito na seção 4.2 calcula o grau de satisfação baseado apenas na localização da residência do usuário.

4.4 Recomendação baseada em Múltiplos Centros Gaussianos

De acordo com o trabalho apresentado em [CML11], existem algumas regiões que os usuários tendem a visitar mais do que outras. No recomendador descrito na seção 4.2, conseguimos explorar uma destas regiões: a região ao redor da residência do usuário. Porém, a idéia de que cada usuário tem apenas uma região de interesse é muito restritiva, pois podem haver outras regiões que despertem interesse no usuário, como por exemplo, a região ao redor de seu local de trabalho.

Para as várias regiões de interesse dos usuários, o modelo de múltiplos centros gaussianos, definido em [CYKL12], propõe aplicar um algoritmo de clusterização que utiliza uma abordagem gulosa. Inicialmente, ordenamos de forma decrescente os POIs de acordo com seu número de visitas. Em seguida, iteramos sobre os POIs e combinamos todos os outros POIs localizados a uma distância menor que d kilômetros do POI da atual iteração, e então esta é uma região candidata. Daí, verificamos se a proporção de check-ins do usuário

nesta região candidata é maior que o *threshold* θ. Caso isso seja verificado, calculamos o centroide da região candidata e assinalamos o mesmo como sendo um centro de interesse para o usuário em questão. Então, seja C_u o conjunto de centros de interesse para um dado usuário u, podemos calcular o grau de preferência de u por uma dada localização l, conforme descrito na equação 4.5.

$$\hat{s}(u,l,c) = \sum_{c_u \in C_u} \frac{1}{dist(l,c_u)} \frac{freq(c_u)^\alpha}{\sum_{c_i \in C_u} freq(c_i)^\alpha} \frac{\delta(l|\mu_{c_u}, \sigma_{c_u})}{\sum_{c_i \in C_u} \delta(l|\mu_{c_i}, \sigma_{c_i})} \tag{4.5}$$

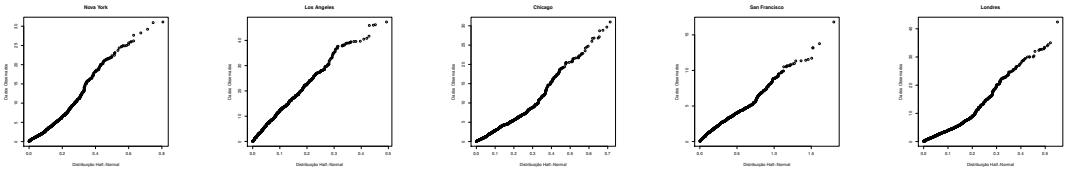
onde a função $freq(c_u)$ retorna o número de check-ins que o usuário u efetuou na região representada pelo centro c_u; $\alpha \in (0, 1]$ é um parâmetro para regular o peso que daremos para a função de frequência; $\delta(l|\mu_{c_u}, \sigma_{c_u})$ é a função de densidade de probabilidade da distribuição gaussiana; e μ_{c_u} e σ_{c_u} correspondem à média e à variância da região ao redor do centro c_u, respectivamente.

4.5 Recomendação baseada em Kernels Gaussianos

Na seção anterior, apresentamos um modelo que infere as regiões de interesse através de uma técnica de clusterização gulosa. Sobretudo para grandes bases de dados, aplicar estes tipos de técnicas pode ser bastante custoso computacionalmente. Na área de aprendizagem de máquina, métodos de aprendizagem baseados em kernels podem ser utilizados para identificar padrões nos dados [STC04]. Sendo assim, nesta seção iremos propor um recomendador baseado em kernels gaussianos que consegue inferir as regiões de interesse dos usuários através de kernels que geramos ao redor de cada check-in.

Para gerar os kernels, utilizamos uma distribuição half-normal, que é um caso especial da distribuição normal onde a média é zero e apenas valores não-negativos são considerados. Julgamos que esta distribuição seria apropriada para gerar recomendações de acordo com as distâncias entre as localizações pelo fato de que o valor máximo da distribuição half-normal é obtido em sua média (zero). Isso implica que se duas localizações forem próximas, ou seja, a distância entre elas é próxima de zero, então o kernel gaussiano irá atribuir a elas uma alta similaridade. À medida em que a distância entre as localizações aumenta, o valor de suas similaridades decai normalmente.

Na figura 4.4 podemos verificar os Q-Q Plots das distâncias dos check-ins efetuados



(a) Nova York (b) Los Angeles (c) Chicago (d) San Francisco (e) Londres

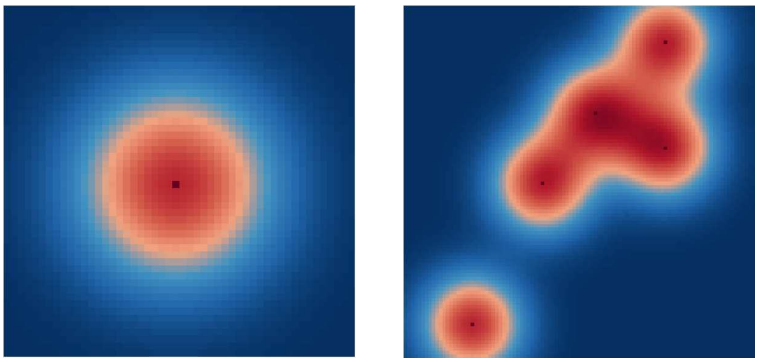
Figura 4.4: Half-Normal Q-Q Plots

pelos usuários e dados de uma distribuição half-normal. Quanto mais próximo a uma linha reta com 45º de inclinação for o gráfico, então maior é a confiança que obtemos de que a distribuição de probabilidades das distâncias dos check-ins é half-normal. Sendo assim, podemos verificar na figura 4.4 que os gráficos dos Q-Q plots das cidades não formam uma reta perfeita com 45º graus na extremidade direita do gráfico, porém na extremidade esquerda (quando as distâncias são menores) podemos ver que seria razoável assumir uma distribuição half-normal. Visto que é esperado que a maioria dos check-ins ocorram a uma distância pequena dos outros locais já visitados pelos usuários anteriormente, então os gráficos da figura 4.4 nos deram maior confiança para implementar uma abordagem baseada em kernels gaussianos.

Sendo assim, o grau de preferência de um usuário u por uma localização l pode ser computado, segundo o recomendador puramente geográfico baseado em kernels gaussianos, da forma descrita na equação 4.6.

$$\hat{s}(u,l,c) = \frac{1}{|L_u|} \sum_{l_u \in L_u} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2} dist(l,l_u)^2} \tag{4.6}$$

Para que fique mais claro como as regiões de interesse dos usuários são inferidas através dos Kernels Gaussianos, considere o exemplo apresentado na figura 4.5, onde cada ponto representa um check-in de um usuário. Na figura 4.5a, podemos ver como o kernel forma uma região de intesse ao redor de um check-in. Neste caso, a cor vermelha representa uma probabilidade mais alta de interesse enquanto a cor azul representa uma probabilidade mais baixa. Na figura 4.5b, podemos ver como os kernels gerados por quatro check-ins próximos se sobrepõem formando uma única região de interesse. Também podemos ver um check-in localizado um pouco mais distante formando uma outra região de interesse menor que a outra região formada pelos quatro check-ins. Sendo assim, retornando ao exemplo descrito



(a) Kernel Gaussiano (b) Interação Entre os Kernels

Figura 4.5: Kernels Gaussianos

na figura 4.1, caso um dado usuário tenha visitado as localizações b e d então, dependendo da distância de c para b e d, a sobreposição dos kernels de b e d poderia levar c a ser recomendado com maior prioridade que a e e.

4.6 Experimentos e Resultados

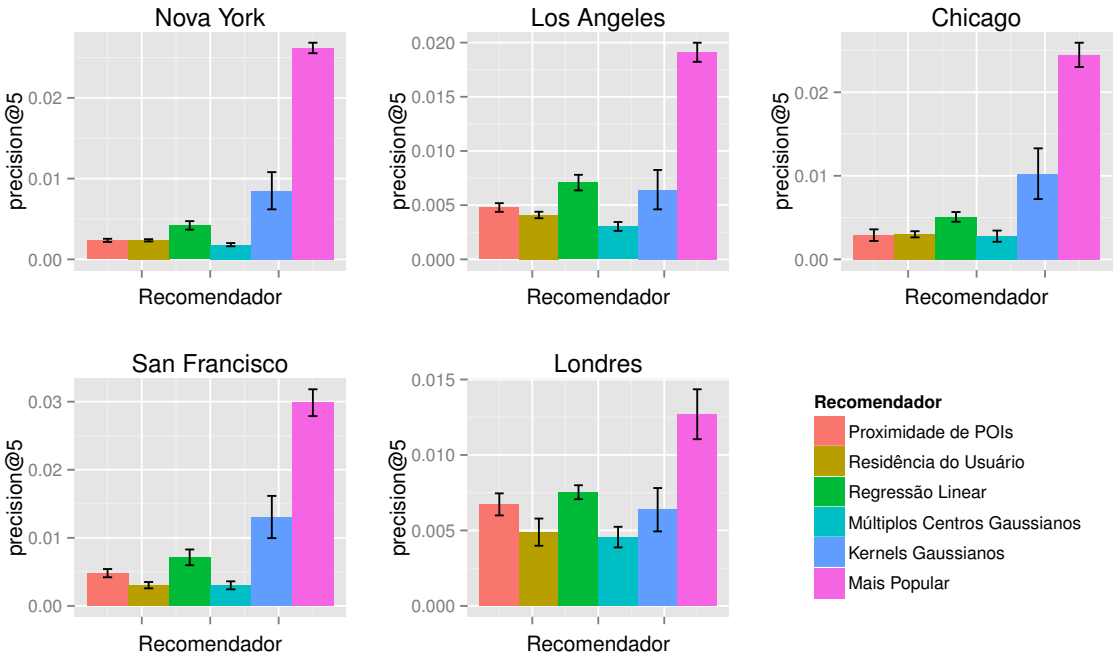
Utilizando a metodologia descrita no capítulo 3, realizamos experimentos para comparar os 5 recomendadores puramente geográficos descritos nas seções anteriores a fim de descobrir quais são os mais acurados. Além dos 5 recomendadores geográficos, utilizamos como baseline o recomendador Mais Popular para ter uma idéia de como é a acurácia destes modelos de uma forma geral. Devido a sua simplicidade, o recomendador Mais Popular é normalmente utilizado como baseline de comparação [KBV09]. No nosso cenário, o recomendador Mais Popular irá ordenar os POIs de acordo com a quantidade de usuários distintos que efetuaram check-ins e recomendar os top-5 melhor ranqueados.

Os resultados dos experimentos estão sintetizados na figura 4.6. Para cada cidade, os recomendadores foram executados sobre os 10 conjuntos de partições de treino e teste, que foram gerados conforme descrito na seção 3.2. Nas figuras 4.6a e 4.6b apresentamos a média e o intervalo de confiança das métricas precision@5 e recall@5. O nível de confiança utilizado para o cálculo dos intervalos foi de 95%.

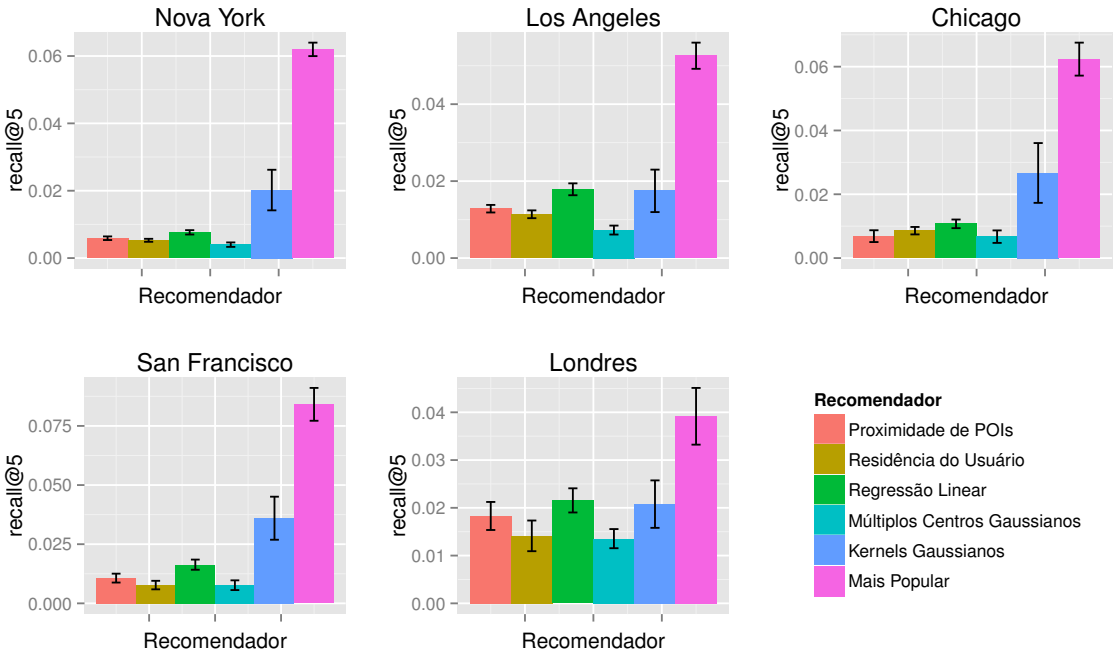
Ao analisar apenas os desempenhos dos recomendadores puramente geográficos, isto é, desconsiderando o recomendador Mais Popular; podemos observar que o recomendador baseado em Kernels Gaussianos alcançou uma maior acurácia na maioria das cidades avali-

adas, mais especificamente em Nova York, Chicago e San Francisco. Nas demais cidades, a acurácia do recomendador baseado em Kernels Gaussianos foi similar a acurácia dos recomendadores baseados em regressão linear e proximidade de POIs. Logo, podemos afirmar com 95% que o recomendador baseado em Kernels Gaussianos apresenta uma acurácia superior ou igual a de todos os outros recomendadores puramente geográficos avaliados.

Ao comparar os resultados dos recomendadores puramente geográficos com o baseline Mais Popular, podemos perceber que este alcançou em todas as cidades avaliadas uma acurácia muito superior. Logo, com 95% de confiança podemos afirmar que o recomendador Mais Popular, apesar de sua simplicidade, consegue ser mais acurado que os recomendadores puramente geográficos. Isso fornece indícios de que os recomendadores não devem utilizar apenas as informações geográficas para gerar recomendações de POIs. Pelo contrário, os recomendadores de POIs devem combinar as informações geográficas com outros modelos de recomendação a fim de obter uma alta acurácia, tal como foi feito em [CYKL12] e [YYLL11].



(a) Precision@5



(b) Recall@5

Figura 4.6: Resultados dos Recomendadores Puramente Geográficos

Capítulo 5

Modelo de Difusão para Recomendação de POIs

Agora que demonstramos que individualmente os modelos de recomendação puramente geográficos não são capazes de alcançar alta acurácia, vamos mostrar neste capítulo como as informações do contexto geográfico podem ser incorporadas em um único modelo baseado em difusão em grafos que também leva em consideração as preferências pessoais dos usuários pelos POIs. Este recomendador foi modelado a partir de três heurísticas acerca do domínio das RSBL que foram obtidas a partir de alguns trabalhos anteriores presentes na literatura.

A primeira heurística que vamos aplicar para o nosso modelo é que usuários que visitaram lugares similares no passado tendem a visitar lugares similares no futuro. Na verdade esta não é uma heurística exclusiva do domínio de RSBL. Esta é a suposição fundamental da técnica de filtragem colaborativa que já foi empregada em vários domínios [KBV09] e que demonstrou bons resultados também no domínio de RSBL [YYLL11].

A segunda heurística é que usuários tendem a visitar lugares próximos daqueles que eles já visitaram no passado. Esta é uma hipótese bem razoável porque é pouco provável que os usuários estejam o tempo todo viajando e visitando lugares muito distantes uns dos outros. Podemos confirmar que esta heurística é aplicável aos nossos dados através da distribuição de probabilidade das distâncias entre os check-ins efetuados pelos usuários, que está descrita na figura 4.3. Como podemos observar, existe uma maior probabilidade de um dado usuário efetuar check-ins em lugares próximos dos lugares nos quais ele já efetuou check-in

anteriormente.

Finalmente a terceira heurística aplicada ao nosso modelo é que os usuários tendem a concentrar seus check-ins em torno de algumas regiões de interesse [CML11; CYKL12]. Para a maioria dos usuários, existem pelo menos dois lugares que eles passam a maior parte do tempo: a sua residência e seu local de trabalho. Logo, existe uma probabilidade maior dos usuários visitarem locais na região que se forma ao redor destes locais. Além destas duas regiões que são comuns, usuários podem apresentar interesse pela região que se forma próxima a residência do(a) namorado(a), um usuário que gosta de esportes provavelmente apresentará interesse pelas regiões formadas perto dos estádios, um usuário que gosta de compras apresentará interesse por regiões com grande concentração de shoppings, e assim sucessivamente.

O resto deste capítulo está organizado da seguinte maneira: Na seção 5.1, descrevemos a fundamentação por trás da técnica de Passeio Aleatório com o intuito de facilitar a compreensão de como a aplicação desta técnica pode ser utilizada para encontrar itens mais relevantes em um grafo. Daí, na seção 5.2 mostraremos como podemos modelar os dados de check-in em um grafo. Em seguida, nas seções 5.3, 5.4 e 5.5 mostramos de forma detalhada como podemos incorporar cada uma destas três heurísticas mencionadas anteriormente, através da definição das arestas do grafo bem como de seus pesos. Finalmente, na seção 5.6 apresentamos como podemos unificar todas as informações em um único modelo de recomendação.

5.1 Passeio Aleatório

Antes de apresentar o nosso modelo de recomendação que utiliza as preferências de usuários por usuários similares, regiões e distâncias; iremos recapitular de forma breve os conceitos fundamentais necessários para a compreensão da técnica de difusão de informações em grafos através de Passeio Aleatório (Random Walk).

Um grafo é uma estrutura de dados definida por um conjunto de vértices (ou nós) e um conjunto de arestas, onde uma aresta conecta dois vértices no grafo. Formalmente, um grafo G pode ser definido como uma tupla $G = (V, E)$ onde V é o conjunto de vértices e $E \subseteq V \times V$ é o conjunto de arestas. Quanto à orientação das arestas, um grafo pode ser

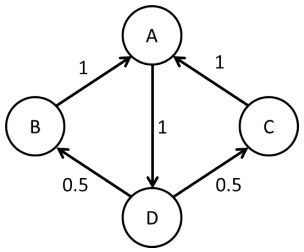


Figura 5.1: Grafo Dirigido e Valorado

classificado como dirigido e não dirigido. Já quanto a ponderação das arestas, um grafo pode ser classificado como valorado ou não valorado. Em um grafo não valorado, considera-se que todas as arestas no grafo têm a mesma importância, ao passo que em um grafo valorado existe uma função $w : E \to \mathbb{R}$ que atribui pesos as arestas para denotar que algumas arestas são mais importantes que outras. Neste caso, quanto maior o peso mais importante é aquela aresta. Na figura 5.1, temos a representação de um grafo dirigido e valorado, onde os círculos representam os vértices, as setas representam as arestas e os números representam os pesos das arestas.

Um grafo pode ser representado através de uma matriz de adjacência $A \in \mathbb{R}^{|V|\times|V|}$. Cada linha e cada coluna em uma matriz de adjacência correspondem a um vértice do grafo. No caso do grafo ser do tipo não valorado, o valor de uma entrada $A_{i,j}$ será igual a 1 se $(i, j) \in E$, e igual a 0 caso contrário. Já se o grafo for do tipo valorado, o valor de cada entrada $A_{i,j}$ será igual ao peso da aresta correspondente, isto é, $A_{i,j} = w(e_{i,j})$. Por exemplo, a matriz de adjacência do grafo da figura 5.1 seria da seguinte forma:

$$A = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0.5 & 0.5 & 0 \end{pmatrix}$$

A idéia por trás do conceito de Passeio Aleatório é a de que se um caminhante começar a pular de nó em nó em um grafo, considerando os pesos das arestas, eventualmente ele irá visitar alguns nós mais frequentemente que outros. Sendo assim, podemos considerar que os nós mais visitados são mais importantes que os nós menos visitados e, portanto, devem ser recomendados com uma maior prioridade. Uma das implementações mais conhecidas

do Passeio Aleatório é o algoritmo de PageRank [PBMW98], que se utiliza da estrutura dos hyperlinks entre páginas web a fim de descobrir quais páginas seriam mais importantes para os usuários em geral; quanto mais páginas tiverem links apontando para uma dada página, mais importante seria esta página.

Os pesos das arestas são de fundamental importância para que o caminhante saiba para que nó ele irá saltar. A probabilidade do caminhante passar por determinada aresta durante um salto é proporcional ao peso daquela aresta. Considere, por exemplo, o grafo da figura 5.1, sempre que o caminhante estiver no vértice A, ele irá saltar para o vértice D. Similarmente, sempre que o caminhante estiver nos vértices B ou C, ele irá saltar para o vértice A. Porém, quando o caminhante estiver no vértice D, ele poderá saltar para o vértice B ou C de forma aleatória. Isso implica que após um número relativamente grande de saltos, é esperado que o caminhante visite os vértices A e D aproximadamente o dobro de vezes que ele visitou os vértices B e C.

Um conceito importante que pode gerar melhores resultados para o Passeio Aleatório é o de teleporte. O teleporte permite ao caminhante pular diretamente para qualquer vértice do grafo, mesmo que não exista uma aresta conectando este vértice com o vértice atual do caminhante. Desta forma, aumentamos a probabilidade do caminhante visitar vértices com poucas (ou até mesmo nenhuma) arestas apontando para ele.

Para que possamos calcular a importância de cada vértice através da técnica de Passeio Aleatório em um grafo G, considere um vetor de probabilidades $\vec{x}$ com $|V|$ dimensões que irá armazenar, a cada instante de tempo (ou iteração), a probabilidade de que o caminhante esteja em cada vértice do grafo. Seja $\vec{x}_t$ os valores armazenados no vetor $\vec{x}$, para uma dada iteração t, esperamos que para um valor suficientemente grande de t, as probabilidades armazenadas no vetor $\vec{x}$ convirjam. Para tanto, a cada iteração iremos atualizar as probabilidades do vetor $\vec{x}$ como descrito na equação 5.1.

$$\vec{x}_{t+1} = \vec{x}_t \left((1-\lambda)A + \frac{\lambda}{|V|} \right) \tag{5.1}$$

Onde A é a matriz de adjacência do grafo e λ é o valor do teleporte. É importante destacar que para garantir a convergência das probabilidades do vetor $\vec{x}$, a matriz A deve apresentar duas propriedades. A primeira delas é que cada entrada da matriz possua valor no intervalo $[0, 1]$. A segunda propriedade é que para cada linha da matriz, a soma das entradas desta

	$\vec{x}[A]$	$\vec{x}[B]$	$\vec{x}[C]$	$\vec{x}[D]$
$\vec{x}_0$	1	0	0	0
$\vec{x}_1$	0.1	0.1	0.1	0.7
$\vec{x}_2$	0.22	0.31	0.31	0.16
$\vec{x}_3$	0.472	0.148	0.148	0.232
...	...	...	...	...
$\vec{x}_{10}$	0.326	0.188	0.188	0.295

Tabela 5.1: Iterações do Passeio Aleatório

linha seja igual a 1. Quando uma matriz apresenta estas duas propriedades, ela é chamada de matriz estocástica ou matriz de Markov.

Como podemos observar, a matriz da transição A para o grafo apresentado na figura 5.1 é uma matriz estocástica. Daí, para aplicar o Passeio Aleatório neste grafo necessitamos inicialmente escolher valores iniciais para o vetor $\vec{x}_0$. Como eventualmente os valores irão convergir, os valores iniciais não têm um grande impacto no resultados final. Então, vamos supor que o caminhante vai começar a caminhada do nó A, logo $\vec{x}_0 = (1, 0, 0, 0)$. Daí, supondo um valor de teleporte $\lambda = 0.4$ e aplicando a equação 5.1, podemos calcular o valor de $\vec{x}_1$ como segue na equação 5.2.

$$\vec{x}_1 = (1, 0, 0, 0) \begin{pmatrix} 0.1 & 0.1 & 0.1 & 0.7 \\ 0.7 & 0.1 & 0.1 & 0.1 \\ 0.7 & 0.1 & 0.1 & 0.1 \\ 01 & 0.4 & 0.4 & 0.1 \end{pmatrix} = (0.1, 0.1, 0.1, 0.7) \tag{5.2}$$

De fato, se continuarmos aplicando sucessivas vezes a equação 5.1, iremos encontrar os valores para o vetor $\vec{x}$ em cada iteração conforme descrito na tabela 5.1. Como podemos observar, em apenas 10 iterações, a técnica de Passeio Aleatório conseguiu detectar corretamente que os vértices A e D são mais importantes que os vértices B e C.

Existem algumas versões da técnica de Passeio Aleatório que utilizam o conceito de Restart. O restart irá permitir que a qualquer iteração o caminhante volte para o vértice onde ele começou a caminhada, mesmo se não existir uma aresta entre o vértice atual e o vértice

inicial. O efeito do restart é diferente do teleporte pelo fato de que o teleporte aumenta a probabilidade de todos os nós de forma uniforme, enquanto o restart insere um viés com relação ao nó inicial.

Como veremos na seção 5.2, no grafo do modelo proposto neste trabalho, teremos um vértice para cada usuário do sistema. Como estamos querendo gerar recomendações personalizadas de acordo com as preferências de um dado usuário alvo, podemos simplesmente assinalar o vértice do usuário alvo como vértice inicial e aplicar o conceito de restart no Passeio Aleatório. Desta forma, estaremos dando mais importância ao nó do usuário alvo e, assim, as recomendações finais terão um viés maior em relação as preferências deste usuário.

5.2 Modelagem dos Dados de Check-in em um Grafo

Para que possamos modelar os dados de check-in em um grafo, é necessário definir inicialmente quais serão os elementos do conjunto de vértices V, bem como os elementos do conjunto de arestas E. Daí, iremos modelar cada usuário e cada POI presente nos dados de check-in como sendo um vértice no grafo. Já o conjunto de arestas pode ser definido da seguinte forma: haverá uma aresta conectando um vértice de usuário para um vértice de um POI sempre que o usuário correspondente tiver efetuado um check-in no POI correspondente.

Formalmente, podemos definir o grafo de feedback de check-ins $G_f = (V_f, E_f)$ como descrito a seguir. Seja U o conjunto de usuários e L o conjunto de lugares presentes no feedback da cidade de interesse, temos $V_f = U \cup L$. Agora, se definirmos uma função booleana checkedIn(u, l) para denotar que um dado usuário $u \in U$ efetuou um check-in em um dado lugar $l \in L$; então poderemos definir o conjunto de arestas do grafo da seguinte forma: $E = \{\{(u, l) : u \in U, l \in L, \text{checkedIn}(u, l) = \mathbf{true}\} \cup \{(l, u) : u \in U, l \in L, \text{checkedIn}(u, l) = \mathbf{true}\}\}$.

Na figura 5.2 podemos encontrar um exemplo de um grafo de feedback de check-ins. Neste caso, temos $U = \{u_1, u_2, u_3\}$ e $L = \{l_1, l_2, l_3, l_4, l_5\}$. Devemos ter sempre em mente que um dos nossos objetivos principais é gerar recomendações personalizadas para um dado usuário alvo. Na figura 5.2 estamos considerando que o usuário alvo é o u_1 e esta é a razão pelo qual o vértice deste usuário está destacado na cor azul escuro. Também podemos observar que os lugares visitados pelo usuário alvo u_1 estão destacadas na cor vermelho. Isso

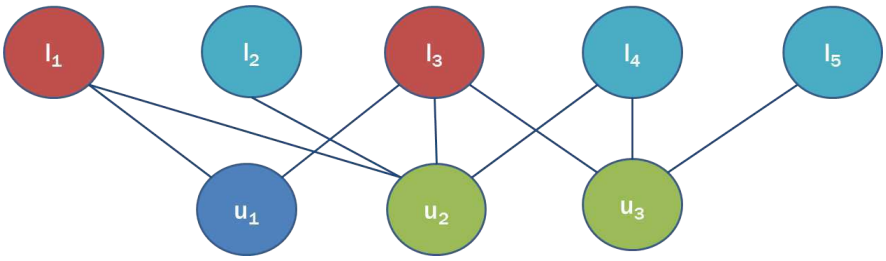


Figura 5.2: Grafo representando os check-ins de usuários em lugares

se dá porque um dos objetivos mais importantes dos sistemas de recomendação é auxiliar os usuários a encontrarem novos itens que possam lhe agradar de acordo com seus gostos pessoais. Portanto, o nosso modelo de recomendação jamais irá recomendar para o usuário alvo lugares que ele já visitou no passado. Logo, no exemplo da figura 5.2 as únicas possíveis localizações que poderiam ser recomendadas para o usuário alvo u_1 seriam l_2, l_4 e l_5.

Se ponderarmos as arestas do grafo de feedback de forma uniforme, isto é, durante as iterações do Passeio Aleatório o caminhante poderá saltar para qualquer nó vizinho com mesma probabilidade, então a tendência é que o caminhante passe mais vezes pelos vértices que possuem uma quantidade maior de arestas de entrada. Como queremos recomendar apenas POIs, então teremos que filtrar o resultados final do Passeio Aleatório para considerar nas recomendações apenas vértices que sejam do tipo POI e que não foram visitados pelo usuário alvo. Sendo assim, se ponderássemos o grafo de forma uniforme, ao executar o Passeio Aleatório e filtrar os resultados; iríamos perceber que os POIs recomendados foram aqueles que têm um maior número de usuários distintos que fizeram check-in. Este resultado é exatamente igual ao que seria gerado por um recomendador que recomenda os itens mais populares do sistema. Formalmente, a ponderação uniforme das arestas do grafo poderia ser feita da seguinte forma: Seja $\text{outDegree}(u) = |\{v \in V : (u, v) \in E\}|$ o número de arestas de saída de um dado vértice $u \in V$, então para ponderar as arestas uniformemente basta ponderar cada uma das arestas do grafo conforme a equação 5.3.

$$w(u, v) = \frac{1}{\text{outDegree}(u)} \tag{5.3}$$

É importante ressaltar que um recomendador do tipo "mais popular" irá sempre gerar as mesmas recomendações independentemente do usuário alvo, que são justamente os itens mais populares do sistema. Tendo em vista que queremos gerar recomendações persona-

lizadas, não iremos considerar este tipo de recomendador no nosso modelo. Na próxima seção, mostraremos como recomendações personalizadas podem ser geradas ao simular os conceitos da filtragem colaborativa dentro da técnica de Passeio Aleatório.

5.3 Difusão Baseada em Filtragem Colaborativa

A principal hipótese das técnicas de recomendação baseadas em filtragem colaborativa é que usuários que apresentaram interesses por itens similares no passado tendem a apresentar interesses por itens similares no futuro. Modelos de recomendação baseados em filtragem colaborativa através da vizinhança de usuários já demonstraram ser bastante eficientes para a recomendação de POIs em RSBL [YYLL11].

Para que um modelo de recomendação baseado nos K-Vizinhos mais próximos seja implementado, é necessário inicialmente definir uma métrica que seja capaz de medir o quão similares são dois usuários. No nosso caso, dois usuários serão mais similares quanto maior for o número de POIs em comum que ambos visitaram no passado. Daí, quanto maior for este número de POIs compartilhados entre dois usuários, mais similares eles devem ser.

Neste trabalho, aplicamos a métrica de similaridade do cosseno para verificar o quão similares são os usuários, pois esta mesma métrica apresentou bons resultados em trabalhos recentes [YYLL11]. Para que esta métrica seja calculada, é necessário representar cada usuário $u \in U$ por meio de um vetor $\vec{u}$ com $|L|$ dimensões. Cada posição do vetor terá valor igual a 1 para indicar que o usuário efetuou um check-in na localização correspondente aquela posição do vetor, ou 0 caso o usuário nunca tenha efetuado um check-in na localização correspondente; tal como modelado em [YYLL11]. Daí, para dois usuários $u, v \in U$, a similaridade do cosseno pode ser calculada como descrito na equação 5.4.

$$sim^{cf}(u, v) = \frac{\vec{u} \cdot \vec{v}}{||\vec{u}||||\vec{v}||} \tag{5.4}$$

Da forma que foi denifida na equação 5.4, os valores das similaridades entre os usuários sempre estarão no intervalo $[0, 1]$. A similaridade entre dois usuários será máxima, isto é igual a 1, quando estes dois usuários tiverem feito check-ins exatamente nos mesmos POIs. Ao passo que a similaridade será mínima, isto é 0, quando os POIs que um usuário fez check-ins são exatamente os POIs que o outro usuário não fez check-in.

Tomando como exemplo os check-ins representados na figura 5.2, teríamos os seguintes vetores para cada um dos usuários: $\vec{u_1} = (1, 0, 1, 0, 0)$, $\vec{u_2} = (1, 1, 1, 1, 0)$ e $\vec{u_3} = (0, 0, 1, 0, 1)$. Como podemos observar, u_1 e u_2 visitaram duas localizações em comum (l_1 e l_3), enquanto u_1 e u_3 visitaram apenas l_3 em comum. Logo, é esperado que as similaridade entre u_1 e u_2 seja maior que a similaridade entre u_1 e u_3. Isso pode ser verificado ao aplicar a equação 5.4. Daí, temos $sim^{cf}(u_1, u_2) = 0.7$ e $sim^{cf}(u_1, u_3) = 0.5$.

Uma vez que podemos calcular a similaridade entre dois usuários, não é difícil encontrar os k usuários mais similares de um dado usuário alvo. Para tanto, basta calcular a similaridade entre o usuário alvo e todos os outros usuários, depois ordenar o resultado em ordem decrescente e por fim selecionar os k primeiros usuários do ranking.

Então, seja N_u o conjunto dos vizinhos (usuários mais similares) de um dado usuário alvo $u \in U$; e $L_{N_u} = \bigcup_{v \in N_u} L_v$ o conjunto de todos os POIs visitados por todos os vizinhos de u. Podemos então definir o grafo G_{cf} o qual a aplicação da técnica de Passeio Aleatório neste grafo irá ranquear as localizações de forma similar à técnica de filtragem colaborativa dos K-Vizinhos mais próximos. Para tanto, devemos adicionar arestas entre u e seus vizinhos N_u, de forma que o caminhante do Passeio Aleatório seja capaz de visitar as localizações L_{N_u}. Formalmente, $G_{cf} = (V_cf, E_cf)$ onde $V_{cf} = \{u\} \cup N_u \cup L_{N_u}$, $E_{cf} = \{(u, v) : v \in N_u\} \cup \{(v, l) : v \in N_u,\, l \in L_{N_u}\} \cup \{(l, v) : l \in L_{N_u},\, v \in N_u\}$ e os pesos das arestas são definidos da seguinte forma:

$$w^{cf}(p, q) = \begin{cases} \frac{sim^{cf}(p,q)}{\sum_{v \in N_u} sim^{cf}(p,v)}, & \text{se } p = u \text{ e } q \in N_u \\ \frac{1}{|L_p|}, & \text{se } p \in N_u \text{ e } q \in L_{N_u} \\ \frac{1}{\text{outDegree}(p)}, & \text{se } p \in L_{N_u} \text{ e } q \in N_u \\ 0, & \text{caso contrário} \end{cases}$$

Observe que os pesos das arestas devem ser normalizados de forma que a soma dos pesos das arestas saindo de cada vértice não seja maior que 1, desta forma os pesos destas arestas podem refletir a probabilidade do caminhante utilizar uma dada aresta durante a execução de um salto no Passeio Aleatório. Na figura 5.3 vemos como seria a disposição dos vértices e arestas ao transformar do grafo de check-ins da figura 1.1 em um grafo baseado em filtragem colaborativa, assumindo u_1 como usuário alvo e u_2 e u_3 como seus dois vizinhos mais

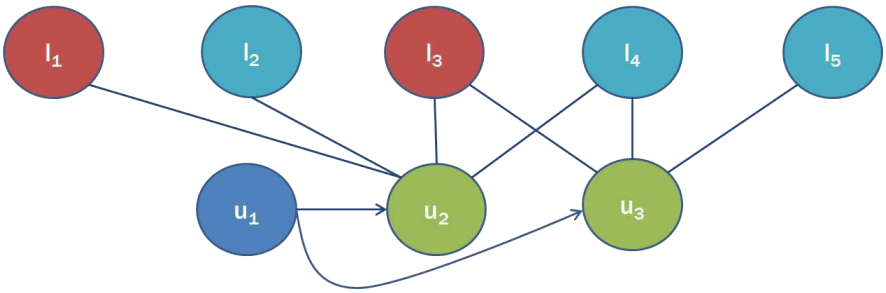


Figura 5.3: Grafo Baseado em Filtragem Colaborativa

próximos.

Quanto aos pesos das arestas do grafo da figura 5.3, sabemos que a similaridades entre o usuário alvo u_1 e seus vizinhos u_2 e u_3 são, respectivamente, 0.7 e 0.5; porém como devemos aplicar normalização, então os pesos das arestas $w^{cf}(u_1, u_2)$ e $w^{cf}(u_1, u_3)$ serão, respectivamente, 0.58 e 0.42. Não apenas as arestas saindo do usuário alvo devem ser normalizadas, mas também as arestas saindo dos vizinhos e saindo dos lugares; por esta razão temos que os pesos de cada aresta saindo do usuário u_2 teriam peso igual a 0.25, enquanto os pesos das arestas chegando em u_2 a partir dos lugares l_1, l_2, l_3 e l_4 seriam 1, 1, 0.5 and 0.5 respectivamente.

Ao aplicar o Passeio Aleatório no grafo da figura 5.3, encontraríamos no ranking de recomendação para o usuário alvo u_1 o lugar l_4 em uma posição melhor que os lugares l_2 e l_5. Isso se dá pois l_4 é alcançável a partir dos dois vizinhos u_2 e u_3, enquanto l_2 é alcançável apenas a partir de u_2 e l_5 apenas a partir de u_3. Assim, o caminhante irá visitar l_4 mais vezes que l_2 e l_5.

5.4 Difusão Baseada em Distâncias

Não é nenhuma surpresa que os check-ins dos usuários nas RSBL não são distribuídos de forma uniforme no espaço geográfico. O deslocamento do usuário de uma cidade para outra, ou até mesmo entre diferentes áreas de uma mesma cidade, geralmente demanda tempo e dinheiro. Por esta razão, os usuários tendem a visitar lugares próximos dos lugares já visitados anteriormente [CCLS11; NSMP11].

Sendo assim, precisamos de uma métrica para definir o quão próximos são dois lugares, sendo a distância geográfica uma escolha óbvia nesse sentido. Porém, desejamos uma

métrica que calcule a similaridade entre lugares no intervalo $[0, 1]$ para manter a coerência com o intervalo da métrica de similaridade entre usuários definida na seção anterior. Sejam $l, l' \in L$ dois lugares os quais queremos medir o quão similares eles são de acordo com sua próximidade geográfica e dist : $L \times L \to \mathbb{R}$ uma função que calcula a distância, em kilômetros, entre dois lugares; então podemos calcular a similaridade entre lugares da RSBL da forma descrita na equação 5.5.

$$sim^{\text{dist}}(l, l') = \min\left(1, \frac{1}{\text{dist}\,(l, l')}\right) \tag{5.5}$$

A função min é usada para manter o cálculo da similaridade no intervalo que desejamos, isto é $[0, 1]$. Caso não tivéssemos aplicado a função min no cálculo da similaridade entre os POIs, obteríamos uma similaridade de valor 2 (fora do intervalo desejado) se dois POIs tivessem distância de 0.5 km, por exemplo. Da forma como definida na equação 5.5, a similaridade entre dois POIs será máxima, isto é 1, se elas distarem até 1 km entre si; caso contrário, a similaridade será menor quanto maior for a distância entre estes POIs.

Agora que temos uma métrica para calcular a similaridade de acordo com a proximidade geográfica entre os POIs, podemos então definir um grafo baseado em distância G_{dist} de forma que, ao aplicar Passeio Aleatório, possamos encontrar os melhores POIs a serem recomendados de acordo apenas com as localizações dos mesmos. Daí, temos $G_{\text{dist}} = (V_{\text{dist}}, E_{\text{dist}})$ onde $V_{\text{dist}} = L$, $E_{\text{dist}} = L \times L$ e os pesos das arestas são calculados de acordo com a equação 5.6.

$$w^{\text{dist}}(p, q) = \frac{sim^{\text{dist}}(p, q)}{\sum_{l \in V_{\text{dist}} \setminus \{p\}} sim^{\text{dist}}(p, l)} \tag{5.6}$$

Observe que a equação 5.6 nada mais é do que uma forma de normalizar as similaridades definidas através da equação 5.5, a fim de garantir que a soma dos pesos das arestas saindo de cada vértice não seja superior a 1.

Na figura 5.4 podemos ver como seria a disposição de vértices e arestas do grafo G_{dist} gerado a partir dos check-ins da figura 5.2. Os pesos das arestas dependem das distâncias entre os vértices, então vamos assumir que geograficamente cada vértice está disposto da mesma forma que na figura 5.4 e que a distância entre dois vértices consecutivos seja igual a 2 km. Isto é, $\text{dist}(l_1, l_2) = \text{dist}(l_2, l_3) = \text{dist}(l_3, l_4) = \text{dist}(l_4, l_5) = 2$ km. Observe que como l_3 é uma localização mais central a média da distância de l_3 para os outros lugares é a menor

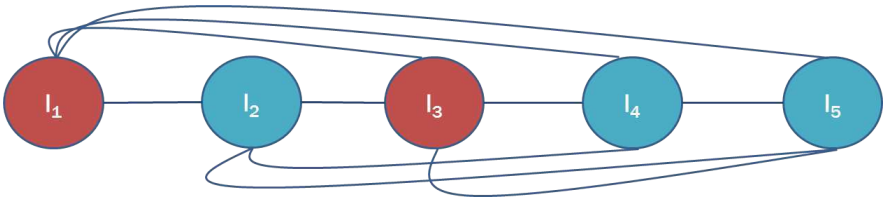


Figura 5.4: Grafo Baseado em Distâncias

entro os cinco, sendo este valor igual a 3 km. Já a distância média dos POIs l_1 e l_5 é igual 5 km pelo fato deles estarem mais distantes do ponto central. Sendo assim, ao aplicar Passeio Aleatório neste grafo, o POI mais visitado seria l_3 pelo fato dele ser em média o lugar mais próximo de todos os outros, porém l_3 está marcado na cor vermelha por já ter sido visitado pelo usuário alvo e, portanto, não ser uma localização apta para a recomendação. Sendo assim, o ranking de recomendação apresentaria l_2 e l_4 nas melhores posições, enquanto l_5 apareceria na última posição por estar em média mais distante de todos os outros lugares.

5.5 Difusão Baseada em Regiões

A distribuição de probabilidade das distâncias entre os check-ins efetuados pelos usuários se aproxima de uma distribuição do tipo lei de potência [CCLS11; NSMP11]. Ou seja, a maioria dos check-ins estão próximos uns dos outros, e esta propriedade foi bem explorada na seção anterior. Porém, o fato de que existem alguns check-ins que são distantes uns dos outro fornecem evidências de que os usuários tendem a eventualmente se deslocarem para áreas distintas.

De fato, os check-ins efetuados pelos usuários das RSBL tipicamente podem ser agrupados em regiões bem definidas [CML11; CYKL12]. Por exemplo, duas regiões que os usuários geralmente efetuam muitos check-ins são a área ao redor de sua residência e a área ao redor do seu local de trabalho.

Para que possamos incorporar as preferências dos usuários por regiões em nosso modelo de recomendação, precisamos inicialmente definir como iremos inferir as diferentes regiões. Para tanto, utilizamos uma abordagem similar à que foi definida em [CML11]: particionamos o mapa mundi em uma grade com células quadradas de mesmo tamanho e então cada célula da grade seria considerada uma região distinta. Apesar dos autores de [CML11] terem

utilizado células de 25 por 25 kilômetros, neste trabalho alcançamos melhores resultados ao diminuir o lado de cada célula para 20 kilômetros. Esta abordagem de particionamento do mundo em grade é semelhante a que foi utilizada para definir as residências dos usuários na seção 4.2. A principal diferença é que para inferir a residência de um dado usuário precisamos descobrir a célula na qual ele efetuou o maior número de check-ins e calcular o centroide destes check-ins. Este cálculo de centroide é desnecessário quando o objetivo do particionamento é apenas o de separar os check-ins dos usuários em diferentes regiões.

Agora que sabemos como inferir as regiões a partir dos dados de check-ins, podemos definir uma métrica para calcular o grau de preferência de um dado usuário por uma dada região. Então seja R o conjunto de regiões, vamos assumir que a preferência de um dado usuário $u \in U$ por uma região $r \in R$ seja proporcional a quantidade de check-ins que u efetuou em r. Daí, podemos formalmente medir a afinidade de um usuário por uma região como descrito na equação 5.7.

$$sim^{\text{reg}}(u, r) = \frac{|L_{u,r}|}{|L_u|} \tag{5.7}$$

onde $L_{u,r}$ é o conjunto de todas os lugares visitados pelo usuário u e que estão localizados dentro da região definida por r. Como podemos observar na equação 5.7, os valores que definem a preferência de um usuário por uma região também estarão dentro do intervalo $[0, 1]$.

Daí, o grafo de difusão baseado em regiões G_{reg} pode ser definido da seguinte forma: $G_{\text{reg}} = (V_{\text{reg}}, E_{\text{reg}})$ onde $V_{\text{reg}} = \{u\} \cup R \cup L$, $E_{\text{reg}} = \{(u, r) : r \in R\} \cup \{(r, l) : r \in R,\, l \in L_r\}$ e os pesos das arestas são definidos da seguinte maneiro:

$$w^{\text{reg}}(p, q) = \begin{cases} sim^{\text{reg}}(p, q), & \text{se } p = u \text{ e } q \in R \\ \frac{1}{|L_p|}, & \text{se } p \in R \text{ e } q \in L_p \\ 0, & \text{caso contrário} \end{cases}$$

onde L_r representa o conjunto de todas as localizações contidas na região $r \in R$. Na figura 5.5, nós estendemos o grafo de check-ins da figura 5.2 com o conceito de regiões. Neste exemplo, inferimos que as localizações l_1 e l_2 pertencem a região r_1, enquanto l_3, l_4 e l_5 pertencem a região r_2. Como podemos ver, o usuário alvo u_1 fez apenas um check-in

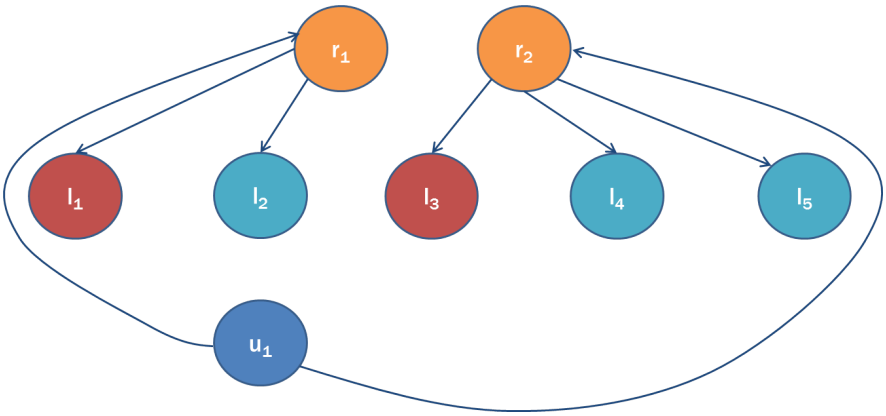


Figura 5.5: Grafo Baseado em Regiões

em cada região, logo ele terá preferências iguais para ambas as regiões. Entretanto, existe apenas um lugar apto para recomendação em r_1 que é l_2, ao passo que existem dois lugares aptos para recomendação em r_2, que são l_4 e l_5.

Daí, ao aplicar o Passeio Aleatório no grafo da figura 5.5, o caminhante visitaria r_1 e r_2 aproximadamente a mesma quantidade de vezes. Porém, uma vez que o caminhante estiver em r_1 ele só poderá saltar para l_2, enquanto ele poderá saltar para l_4 ou l_5 sempre que estiver em r_2. Por esta razão, de acordo com a difusão baseada em regiões, o POI l_2 seria ranqueado em uma posição mais alta que os POIs l_4 e l_5.

5.6 Modelo Unificado de Difusão

Nesta seção, demonstraremos como combinar os três modelos apresentados nas seções anteriores de modo a gerar um único modelo de difusão em grafos capaz de capturar, de forma transparente, as preferências dos usuários por POIs, distâncias e regiões. Sendo assim, seja $u \in U$ o usuário alvo ao qual desejamos recomendar novos lugares para que ele possa visitar, podemos definir o grafo unificado de difusão G_{unif} da seguinte forma: $G_{\text{unif}} = (V_{\text{unif}}, E_{\text{unif}})$ onde $V_{\text{unif}} = V_{cf} \cup V_{\text{dist}} \cup V_{\text{reg}}$, $E_{\text{unif}} = E_{cf} \cup E_{\text{dist}} \cup E_{\text{reg}}$ e os pesos de suas arestas são definidos da seguinte forma:

$$w^{\text{unif}}(p,q) = \begin{cases} \alpha w^{cf}(p,q), & \text{se } p = u \text{ e } q \in N_u \\ \beta w^{cf}(p,q), & \text{se } p \in N_u \text{ e } q \in L_{N_u} \\ \gamma w^{cf}(p,q), & \text{se } p \in L_{N_u} \text{ e } q \in N_u \\ \delta w^{\text{dist}}(p,q), & \text{se } \{p,q\} \subseteq L_{N_u} \cup L_u \\ \theta w^{reg}(p,q), & \text{se } p = u \text{ e } q \in R \\ w^{\text{reg}}(p,q), & \text{se } p \in R \text{ e } q \in L_p \\ 0, & \text{caso contrário} \end{cases}$$

Para que a técnica de Passeio Aleatório seja executada corretamente, é importante que a soma dos pesos das arestas que saem de qualquer vértice seja maior que 1. Logo, os valores dos hiperparâmetros $\alpha, \beta, \gamma, \delta$ e θ não devem ser superiores a 1. Além disso, visto que α e θ ponderam arestas que se originam no usuário alvo u, então a soma $\alpha + \theta$ não deve ser superior a 1. Similarmente, γ e δ ponderam arestas que se originam em lugares, logo a soma $\gamma + \delta$ também não deve ser superior a 1.

Na figura 5.6, podemos ver como seria o grafo unificado considerando os check-ins modelados na figura 5.6 e as regiões inferidas no modelo de difusão baseado em regiões da figura 5.5. Por questões de legibilidade, omitimos algumas arestas do modelo baseado em distâncias da figura 5.4.

Para gerar as recomendações a partir do modelo de difusão unificado, basta efetuar o Passeio Aleatório sob o grafo unificado. Após a convergência do Passeio Aleatório, ordenamos os POIs em ordem descrescente de probabilidade e por fim selecionamos os 5 primeiros POIs da lista para a recomendar. Também devemos destacar que diferentemente de outros trabalhos [CYKL12; YYLL11], nós não estamos gerando e combinando dois ou mais rankings de POIs a partir de modelos distintos. O nosso ranking é gerado por um único modelo que consegue incorporar todas as três heurísticas assumidas no início deste capítulo.

No algoritmo 1 encontramos o pseudocódigo para gerar recomendações para um dado usuário u uma vez que seu grafo unificado G_{unif} tenha sido calculado. Existem outros três parâmetros importantes para gerar a recomendação. A probabilidade de teleporte λ e a probabilidade de restart influenciam os saltos do caminhante durante a execução do Passeio Aleatório. Já o threshold é um valor utilizado como critério de parada do Passeio Aleatório,

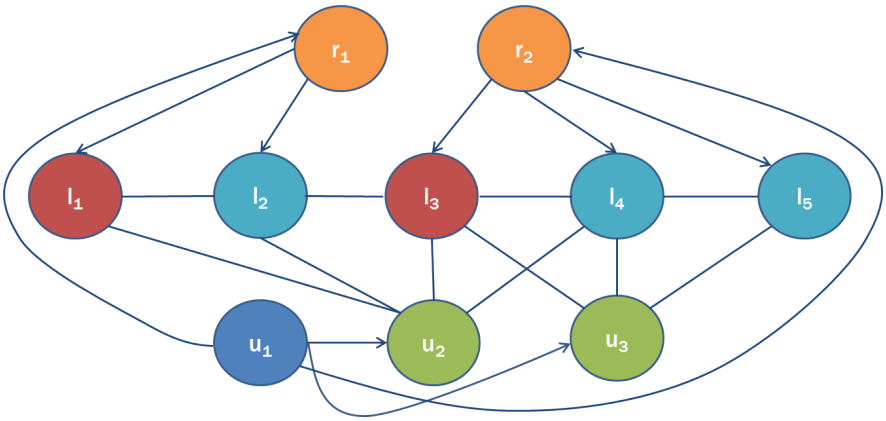


Figura 5.6: Grafo Unificado

isto é, este parâmetro indica que o vetor que armazena as probabilidades do caminhante estar em cada um dos vértices do gráfico atingiu um estado estável. Nos nossos experimentos utilizamos os seguintes valores, que foram definidos através de validação cruzada, para cada um destes três parâmetros: $\lambda = 0.1$, probabilidade de restart = 0.01 e threshold = 0.99.

Entre as linhas 2 e 6 do algoritmo 1, criamos um vetor com $|V_{\text{unif}}|$ dimensões. Cada elemento do vetor corresponde a um vértice do grafo. Sendo assim, inicializamos o elemento correspondente ao vértice do usuário alvo u com o valor 1, e os demais elementos do vetor com o valor 0. Entre as linhas 7 e 10, configuramos a matriz de transição de probabilidades do grafo G_{unif} para que a mesma leve em consideração as probabilidades de teleporte e restart do caminhante. Em seguida na linha 11, assinalamos o vetor a ser usado na primeira iteração do passei aleatório como sendo o vetor inicializado nas linhas de 2 a 6. Isso implica que o caminhante sempre irá começar o passeio aleatório pelo usuário alvo, pois no vetor inicial a probabilidade deste vértice é 1 enquanto as probabilidades dos demais vértices do grafo é 0. Entra as linhas 12 e 15, executamos o passeio aleatório até que as probabilidades do vetor dos vértices alcance um estado estável. Entre as linhas 16 e 18, realizamos uma filtragem no vetor de probabilidade dos vértices, pois somente iremos recomendar POIs. Logo, podemos desconsiderar todos os vértices do tipo usuário ou região. Em seguida, ordenamos de forma descrescente os POIs de acordo com suas probabilidades finais. Finalmente, recomendamos os 5 POIs melhores ranqueados.

Seja $n = |V_{\text{unif}}|$ a quantidade de vértices do grafo unificado então a complexidade do algortimo 1 para gerar recomendações para um dado usuário alvo u é de $O(n^2 + n * logn + n)$.

Algorithm 1 Recomendação de POIs através do Passeio Aleatório

1: **Recomendar**$(u, G_{\text{unif}}, \lambda, restart, threshold)$
2: $initialVector =$ New Vector($|V_{\text{unif}}|$)
3: userIndex = initialVector.indexOf(u)
4: **for** $i = 1$ to $length(initialVector)$ **do**
5: v[i] = (i == userIndex) ? 1 : 0
6: **end for**
7: A = getTrasitionMatrix(G_{unif})
8: A = (1-restart) * A
9: A[,userIndex] = A[,userIndex] + restart
10: A = (1 - λ)A + (λ / $|V_{\text{unif}}|$)
11: vector = initialVector
12: **repeat**
13: oldVector = vector
14: vector = vector * A
15: **until** cossine(vector, oldVector) $<=$ threshold
16: poiVector = vector.filterByType(POI)
17: sort(poiVector, order.descending)
18: **return** poiVector[1:5]

Isso se dá pois existem três operações que precisam serem efetuadas para a recomendação: o Passeio Aleatório, a ordenação dos vértices e a filtragem dos vértices. Para executar o Passeio Aleatório precisamos a cada iteração multiplicar um vetor de tamanho n com uma matriz quadrada de dimensões $n \times n$, e o custo desta operação é $O(n^2)$. A ordenação dos vértices em ordem decrescente é uma operação com complexidade $O(n * logn)$ se a mesma for implementada com um algortimo eficiente, como o merge sort por exemplo. Por fim, a operação de filtragem dos vértices do tipo POI tem complexidade $O(n)$.

Capítulo 6

Validação do Modelo de Difusão

Neste capítulo, descrevemos como validamos o modelo de recomendação de POIs baseado em difusão em grafos descrito na seção 5.6. Nos referiremos a este modelo de agora em diante pela sigla DGM (Diffusion Geographic Model). Os valores dos hiperparâmetros do nosso modelo utilizados nos experimentos foram: $k = 80$ (o número de vizinhos mais próximos), $\alpha = 0.5$, $\beta = 0.25$, $\gamma = 0.9$, $\delta = 0.1$ e $\theta = 0.25$. Estes valores foram definidos através de validação cruzada. É importante destacar que o protocolo de avaliação, a descrição dos dados e das métricas utilizadas nesta validação foram apresentados no capítulo 3.

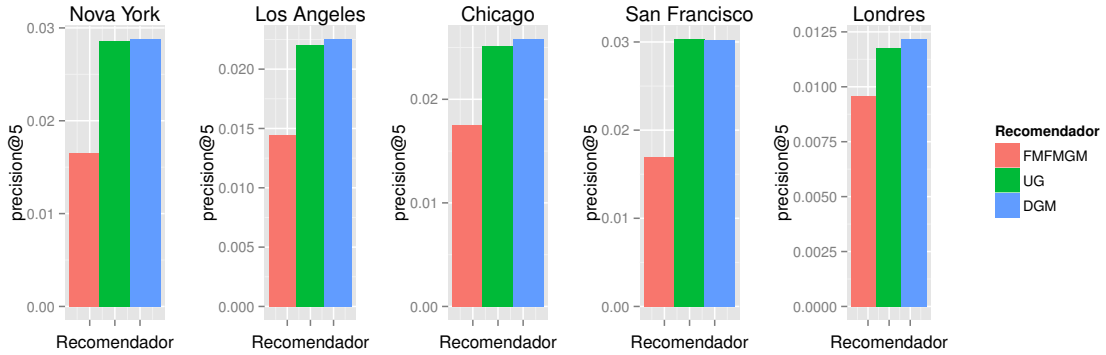
O resto deste capítulo está organizado da seguinte maneira: Na seção 6.1, descrevemos os recomendadores de POIs presentes no estado-da-arte que serão utilizados como abordagens comparativas para a validação do recomendador DGM. Em seguida, na seção 6.2 apresentamos os resultados dos experimentos realizados para comparar os recomendadores de POIs. Em nossa análise pudemos verificar que o recomendador DGM apresenta acurácia igual ou superior aos demais recomendadores em todas as cidades avaliadas. Finalmente, na seção 6.3 apresentamos os experimentos realizados para comparar o desempenho dos recomendadores de POIs. Em nossa análise pudemos constatar que o recomendador DGM é mais eficiente que os demais recomendadores de POIs em todas as cidades avaliadas.

6.1 Abordagens Comparativas

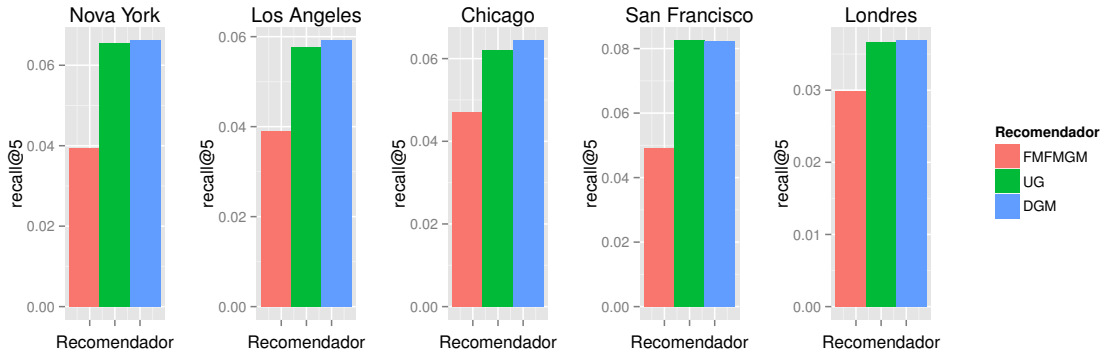
Nós comparamos nosso modelo de difusão para recomendação de POIs com dois modelos de recomendação presentes no estado-da-arte, que foram apresentados em [CYKL12] e [YYLL11]. Ambas abordagens utilizam uma combinação de modelos de filtragem colaborativa, modelos cientes de contexto geográfico e modelos cientes de contexto social. Neste trabalho, iremos considerar apenas os componentes responsáveis por modelar as técnicas de filtragem colaborativa e os modelos cientes de contexto geográfico. Desconsideramos, então, os modelos cientes de contexto social. Existem duas razões principais que nos levaram a tomar esta decisão: a primeira é o fato de que a base de dados do Foursquare que usamos não continha as relações de amizade entre os usuários; a segunda razão é que de acordo com Ye et al. [YYLL11], o cenário onde a relação social exerce uma maior influência sobre os check-ins dos usuário é quando os usuários estão viajando para cidades distantes, e este não é o cenário que estamos explorando neste trabalho. Nós iremos descrever de forma breve cada uma das duas abordagens a seguir.

O recomendador geográfico utilizado por Cheng et al. [CYKL12] foi o modelo baseado em múltiplos centros gaussianos descrito na seção 4.4. Para o componente baseado em filtragem colaborativa, foi utilizado um modelo de fatoração proabilística de matrizes [SM08] que leva em consideração a frequência de check-ins dos usuários. É importante destacar que a combinação destes dois modelos foi feita utilizando uma simples multiplicação. Nós vamos nos referir a esta abordagem pela sigla FMFMGM (Fused Matrix Factorization framework with the Multi-center Gaussian Model). Os valores dos hiperparâmetros do modelo geográfico do FMFMGM utilizados em nossos experimentos foram: $d = 30$, $\alpha = 0.2$, $\theta = 0.02$. Já os valores dos hiperparâmetros do modelo de fatoração de matrizes do FMFMGM foram: $\alpha = 20$, $\beta = 0.2$, $\lambda = 0.001$ and $k = 10$. Estes valores foram definidos através de validação cruzada.

O recomendador geográfico utilizado por Ye et al. [YYLL11] foi o modelo baseado em regressão linear descrito na seção 4.3. Para o componente baseado em filtragem colaborativa, os autores utilizaram a técnica dos k-vizinhos mais próximo com a métrica do cosseno sendo usada como função de similaridade entre os usuários. Estes dois modelos foram integrados através de uma simples combinação linear. Nos nossos experimentos, nós atribuímos



(a) Precision@5



(b) Recall@5

Figura 6.1: Resultados da Acurácia dos Recomendadores

o peso de 0.05 para o componente geográfico e 0.95 para o componente baseado em filtragem colaborativa. Estes valores também foram definidos através de validação cruzada. De forma similar ao trabalho original, nós vamos nos referir a este modelo pela sigla UG para denotar que ele combina as preferências entre os usuários (U) e as influências geográficas (G). Implementamos cada uma destas abordagens comparativas seguindo, com o máximo de fidelidade possível, suas descrições nos trabalhos em que foram apresentadas.

6.2 Avaliação da Acurácia

Os resultados dos experimentos comparando a acurácia dos recomendadores estão sintetizados na figura 6.1. Podemos observar nas figuras 6.1a e 6.1b a média das métricas precision@5 e recall@5 após os recomendadores serem executados sobre os 10 conjuntos de partições de treino e teste de cada cidade, que foram gerados conforme descrito na seção 3.2.

Podemos observar que a abordagem proposta neste trabalho apresenta uma melhor acurácia do que as outras abordagens na maioria das cidades avaliadas de acordo com ambas as métricas, precision@5 e recall@5. Porém, é importante destacar que enquanto as abordagens UG e DGM apresentaram um desempenho similar, a abordagem FMFMGM apresentou uma acurácia bastante inferior as outras duas. Isso pode ser explicado pelo fato de que o componente geográfico da abordagem FMFMGM isoladamente apresenta um desempenho inferior que o componente geográfico da abordagem UG, como vimos nos resultados demonstrados na seção 4.6. Além disso, como visto na tabela 3.1, o nível de esparsidade das nossas bases de dados é muito elevado, o que pode ter dificultado o aprendizado do componente baseado em fatoração de matrizes da abordagem FMFMGM.

Apesar da diferença entre as abordagens UG e DGM parecer pequena, esta diferença é estatisticamente significativa em alguns casos. Nos gráficos da figura 6.1, observamos apenas a média de cada métrica após a execução dos modelos em 10 partições para cada cidade, porém quando olhamos os valores pareados de cada partição, iremos perceber para algumas cidades uma superioridade na acurácia da abordagem DGM. Por exemplo, considere os resultados da métrica precision@5 em cada uma das 10 partições na cidade de chicago, descritos na tabela 6.1. Nesta tabela, podemos ver claramente que a acurácia da abordagem UG se igualou à acurácia do recomendador DGM em apenas duas partições e foi inferior nas outras 8.

Sendo assim, para verificarmos se a diferença entre as acurácia das abordagens UG e DGM era significativa estatisticamente, decidimos aplicar o teste t de Student pareado. Este teste consiste em verificar se a diferença entre duas amostras pareadas é igual, maior ou menor que zero com um certo nível de confiança. Para as nossas análises utilizamos um nível de 95% de confiança. Daí, escolhemos a seguinte hipótese nula para o teste: A acurácia do recomendador UG é maior ou igual do que a acurácia do DGM. Assim a hipótese alternativa seria: A acurária do recomendador UG é menor que a do DGM. A saída de um teste de hipótese é um p-valor. Como queremos um nível de confiança de 95%, então se o p-valor for menor que $0.05(1 - 95\%)$ então podemos rejeitar a hipótese nula e aceitar a hipótese alternativa como verdadeira. Neste caso, se o p-valor das nossas análises for menor que 0.05, podemos concluir que a acurácia de UG é pior que a de DGM; caso contrário a acurária de UG pode ser maior ou igual a de DGM.

Partição	UG	DGM	UG - DGM
01	0,0244617	0,0260120	-0,00155030
02	0,02153318	0,02153318	0,00000000
03	0,02532303	0,02601209	-0,00068906
04	0,02049959	0,02118865	-0,00068906
05	0,02859608	0,02876834	-0,00017226
06	0,02480623	0,02515076	-0,00034453
07	0,02756249	0,02773475	-0,00017226
08	0,02807928	0,02894061	-0,00086133
09	0,02549529	0,02773475	-0,00223946
10	0,02497849	0,02497849	0,00000000
Média	0,02513354	0,02580536	-0,000671826

Tabela 6.1: Precision@5 na cidade de Chicago

Cidade	Nova York	Los Angeles	Chicago	San Francisco	Londres
Precision@5	$0,122$	$0,0006$	$0,008$	$0,646$	$0,103$
Recall@5	$0,130$	$0,02019$	$0,006$	$0,628$	$0,424$

Tabela 6.2: Teste T de Student Pareado - Hipótese Nula: UG $>=$ DGM

Os resultados dos testes de hipótese para cada cidade referentes a cada umas das métricas avaliadas estão descritos na tabela 6.2. Como era esperado pelos dados demonstrados na tabela 6.1, o p-valor para a métrica Precision@5 na cidade de Chicago foi menor que 0.05, o que nos faz aceitar a hipótese alternativa como verdadeira, ou seja, neste caso a acurácia do recomendador DGM foi maior que a do UG com 95% de confiança. De fato, ao analisar os resultados da tabela 6.2 percebemos que na cidade de Los Angeles a hipótese nula também foi rejeitada.

As três cidades onde a hipótese nula não foi rejeitada foram Nova York, San Francisco e Londres. Nestes casos, podemos concluir apenas que ou a acurácia de UG é maior que DGM, ou ambas abordagens apresentam acurácias estatisticamente iguais. Então efetuamos um novo teste t de Student pareado, sendo que desta vez as hipóteses nulas e alternativas foram

Cidade	Nova York	San Francisco	Londres
Precision@5	$0,245$	$0,707$	$0,206$
Recall@5	$0,260$	$0,742$	$0,849$

Tabela 6.3: Teste T de Student Pareado - Hipótese Nula: UG = DGM

ligeiramente diferentes. Agora, a hipótese nula seria: UG e DGM apresentam acurácias iguais. Logo a hipótese alternativa seria: UG e DGM apresentam acurácias diferentes. Os p-valores encontrados para estas três cidades estão descritos na tabela 6.3. Isso implica que para os três casos aceitamos a hipótese nula de que as acurácias são iguais.

Portanto, de acordo com os resultados de nossos experimentos e análises, podemos concluir com 95% de confiança que o recomendador DGM baseado em difusão em grafos apresentou uma acurácia superior a todas as abordagens comparativas nas cidades de Los Angeles e Chicago. Em Nova York, San Francisco e Londres o mesmo apresentou acurácia equivalente a abordagem UG. Em todas as cidades a acurácia do recomendador DGM foi superior a acurácia do modelo FMFMGM. Logo, podemos concluir com 95% de confiança que o recomendador DGM apresenta acurácia igual ou superior se comparado com os outros recomendadores de POIs presentes no estado-da-arte.

6.3 Avaliação do Desempenho

O recomendador DGM tem entre suas principais vantagens o fato de conseguir agregar em um único modelo de difusão informações a respeito das similaridades entre os usuários e informações do contexto geográfico. Pelo fato de que os recomendadores de POIs presentes no estado-da-arte precisam executar mais de um modelo separado para gerar as recomendações para os usuários, é esperado que o recomendador DGM apresente um desempenho mais eficiente. Nesta seção, iremos apresentar os resultados dos experimentos comparando os desempenhos do recomendador DGM e das abordagens comparativas, que confirmam a intuição de que o recomendador DGM é mais eficiente que os demais.

A métrica utilizada para medir o desempenho dos recomendadores foi o tempo, em milissegundos. O tempo calculado foi o intervalo necessário para treinar os recomendadores e gerar uma lista de recomendações para cada usuário de cada cidade. Este cálculo foi efe-

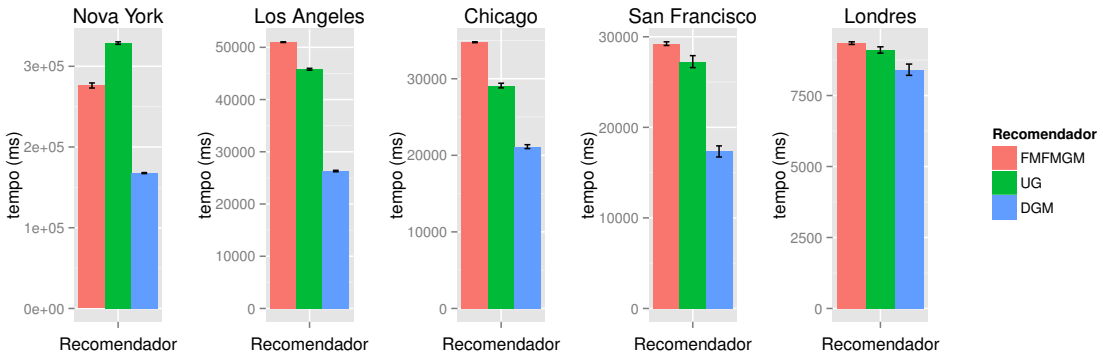


Figura 6.2: Resultados do Desempenho dos Recomendadores

tuado para cada partição de treino e teste de cada cidade. Os experimentos para calcular o tempo de execução dos recomendadores foram efetuados em um computador com as seguintes configurações: processador Intel Xeon 3.30GHz, 8GB de memória RAM e sistema operacional Windows 7. A média dos tempos necessários para que cada recomendador consiga gerar as recomendações, juntamente com seus intervalos de confiança, estão apresentados na figura 6.2. O nível de confiança dos intervalos é de 95%.

Como podemos observar na figura 6.2, o recomendador DGM desempenho superior aos demais recomendadores em todas as cidades avaliadas. Logo, podemos afirmar com 95% de confiança que o modelo proposto neste trabalho consegue manter acurácia igual ou superior aos recomendadores de POIs presentes no estado-da-arte, porém seu desempenho é superior ao dos recomendadores de POIs presentes no estado-da-arte.

Capítulo 7

Conclusões

Neste trabalho, nós lidamos com o problema de recomendação de POIs no contexto de RSBL quando sabemos com precisão a cidade na qual o usuário alvo da recomendação está localizado. Para tanto, inicialmente fizemos uma análise a fim de verificar a real contribuição que o contexto geográfico pode fornecer de forma isolada para os recomendadores. Propomos um novo recomendador geográfico baseado em Kernels Gaussianos, que consegue alcançar acurácia igual ou superior aos demais recomendadores puramente geográficos presentes no estado-da-arte. Porém, também verificamos que isoladamente os recomendadores puramente geográficos não são capazes de alcançar alta acurácia. Logo, faz-se necessário combinar as informações deste contexto com outras informações.

Diferentemente dos trabalhos já apresentados na literatura que modelavam cada aspecto do domínio em um recomendador distinto, neste trabalho nós incorporamos vários tipos de preferências em um único modelo baseado em difusão em grafos. Estas preferências levaram em conta as similaridades entre as preferências dos usuários, a influência exercida pelas distâncias entre os POIs e as preferências que os usuários têm por determinadas regiões. Através de um conjunto de experimentos utilizando dados reais de uma das RSBL mais populares atualmente (Foursquare), demonstramos que a abordagem proposta apresenta uma acurácia igual ou maior que as abordagens definidas no estado-da-arte nas cidades avaliadas. Também verificamos que o modelo proposto apresenta um desempenho superior do que as abordagens definidas no estado-da-arte. Logo, o modelo proposto consegue ser mais eficiente alcançado uma acurácia igual ou melhor que os recomendadores de POIs existentes do estado-da-arte.

Como o modelo proposto é flexível no sentido de incorporar novos contextos, então para trabalhos futuros pretendemos incorporar mais informações ao grafo. Entre os contextos que acreditamos serem promissores para melhorar a acurácia do recomendador proposto, podemos citar o contexto das categorias dos POIs e o contexto temporal dos check-ins. Apesar destes contextos já terem sido explorados na literatura de forma isolada [GTHL13; ZM12; LLAM13], eles ainda não foram incorporados em um único modelo de difusão, tal como foi proposto neste trabalho.

Bibliografia

[BL11] Lars Backstrom and Jure Leskovec. Supervised random walks: Predicting and recommending links in social networks. In *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*, WSDM '11, pages 635–644, New York, NY, USA, 2011. ACM.

[CCLS11] Zhiyuan Cheng, James Caverlee, Kyumin Lee, and Daniel Z. Sui. Exploring millions of footprints in location sharing services. In Lada A. Adamic, Ricardo A. Baeza-Yates, and Scott Counts, editors, *ICWSM*. The AAAI Press, 2011.

[CML11] Eunjoon Cho, Seth A Myers, and Jure Leskovec. Friendship and mobility: user movement in location-based social networks. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1082–1090. ACM, 2011.

[CYKL12] Chen Cheng, Haiqin Yang, Irwin King, and Michael R. Lyu. Fused matrix factorization with geographical and social influence in location-based social networks. In *AAAI*, 2012.

[DA00] Anind K. Dey and Gregory D. Abowd. Towards a better understanding of context and context-awareness. In *Workshop on The What, Who, Where, When, and How of Context-Awareness*, 2000 Conference on Human Factors in Computing Systems, 2000.

[GTHL13] Huiji Gao, Jiliang Tang, Xia Hu, and Huan Liu. Exploring temporal effects for location recommendation on location-based social networks. In Qiang Yang

0001, Irwin King, Qing Li, Pearl Pu, and George Karypis, editors, *RecSys*, pages 93–100. ACM, 2013.

[HE13] Bo Hu and Martin Ester. Spatial topic modeling in online social media for location recommendation. In Qiang Yang 0001, Irwin King, Qing Li, Pearl Pu, and George Karypis, editors, *RecSys*, pages 25–32. ACM, 2013.

[JMH+08] Robert Jäschke, Leandro Marinho, Andreas Hotho, Lars Schmidt-Thieme, and Gerd Stumme. Tag recommendations in social bookmarking systems. *AI Commun.*, 21(4):231–247, December 2008.

[KBV09] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

[KIH+13] Takeshi Kurashima, Tomoharu Iwata, Takahide Hoshide, Noriko Takaya, and Ko Fujimura. Geo topic model: Joint modeling of user's activity area and interests for location recommendation. In *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining*, WSDM '13, pages 375–384, New York, NY, USA, 2013. ACM.

[KSJ09] Ioannis Konstas, Vassilios Stathopoulos, and Joemon M. Jose. On social networks and collaborative recommendation. In *Proceedings of the 32Nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '09, pages 195–202, New York, NY, USA, 2009. ACM.

[LLAM13] Xin Liu, Yong Liu, Karl Aberer, and Chunyan Miao. Personalized point-of-interest recommendation by mining users' preference transition. In Qi He, Arun Iyengar, Wolfgang Nejdl, Jian Pei, and Rajeev Rastogi, editors, *CIKM*, pages 733–738. ACM, 2013.

[NSMP11] Anastasios Noulas, Salvatore Scellato, Cecilia Mascolo, and Massimiliano Pontil. An empirical study of geographic user activity patterns in foursquare. In Lada A. Adamic, Ricardo A. Baeza-Yates, and Scott Counts, editors, *ICWSM*. The AAAI Press, 2011.

[PBMW98] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. In *Proceedings of the 7th International World Wide Web Conference*, pages 161–172, Brisbane, Australia, 1998.

[SdMA+14] Thiago H. Silva, Pedro O. S. Vaz de Melo, Jussara M. Almeida, Mirco Musolesi, and Antonio A. F. Loureiro. You are what you eat (and drink): Identifying cultural boundaries by analyzing food & drink habits in foursquare. *CoRR*, abs/1404.1009, 2014.

[SM08] Ruslan Salakhutdinov and Andriy Mnih. Probabilistic matrix factorization. In *Advances in Neural Information Processing Systems*, volume 20, 2008.

[STC04] John Shawe-Taylor and Nello Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, New York, NY, USA, 2004.

[YSC+13] Hongzhi Yin, Yizhou Sun, Bin Cui, Zhiting Hu, and Ling Chen. Lcars: a location-content-aware recommender system. In Inderjit S. Dhillon, Yehuda Koren, Rayid Ghani, Ted E. Senator, Paul Bradley, Rajesh Parekh, Jingrui He, Robert L. Grossman, and Ramasamy Uthurusamy, editors, *KDD*, pages 221–229. ACM, 2013.

[YYLL11] Mao Ye, Peifeng Yin, Wang-Chien Lee, and Dik Lun Lee. Exploiting geographical influence for collaborative point-of-interest recommendation. In Wei-Ying Ma, Jian-Yun Nie, Ricardo A. Baeza-Yates, Tat-Seng Chua, and W. Bruce Croft, editors, *SIGIR*, pages 325–334. ACM, 2011.

[ZM12] Jie Bao 0003, Yu Zheng, and Mohamed F. Mokbel. Location-based and preference-aware recommendation using sparse geo-social networking data. In Isabel F. Cruz, Craig Knoblock, Peer Kröger, Egemen Tanin, and Peter Widmayer, editors, *SIGSPATIAL/GIS*, pages 199–208. ACM, 2012.