



**UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

NÍCOLAS MOREIRA NOBRE LEITE

**AVALIAÇÃO DE EMBEDDINGS GEOGRÁFICOS EM SISTEMAS
DE RECOMENDAÇÃO DE POIS**

CAMPINA GRANDE - PB

2024

NÍCOLAS MOREIRA NOBRE LEITE

**AVALIAÇÃO DE EMBEDDINGS GEOGRÁFICOS EM SISTEMAS
DE RECOMENDAÇÃO DE POIS**

**Trabalho de Conclusão Curso
apresentado ao Curso Bacharelado em
Ciência da Computação do Centro de
Engenharia Elétrica e Informática da
Universidade Federal de Campina
Grande, como requisito parcial para
obtenção do título de Bacharel em
Ciência da Computação.**

Orientador : Cláudio Elízio Calazans Campelo

CAMPINA GRANDE - PB

2024

NÍCOLAS MOREIRA NOBRE LEITE

**AVALIAÇÃO DE EMBEDDINGS GEOGRÁFICOS EM SISTEMAS
DE RECOMENDAÇÃO DE POIS**

**Trabalho de Conclusão Curso
apresentado ao Curso Bacharelado em
Ciência da Computação do Centro de
Engenharia Elétrica e Informática da
Universidade Federal de Campina
Grande, como requisito parcial para
obtenção do título de Bacharel em
Ciência da Computação.**

BANCA EXAMINADORA:

**Cláudio Elízio Calazans Campelo
Orientador – UASC/CEEI/UFCG**

**Carlos Eduardo Santos Pires
Examinador – UASC/CEEI/UFCG**

**Francisco Vilar Brasileiro
Professor da Disciplina TCC – UASC/CEEI/UFCG**

Trabalho aprovado em 15 de Maio de 2024.

CAMPINA GRANDE - PB

RESUMO

A recomendação de Pontos de Interesse (POIs) ganha destaque no contexto de sistemas de recomendação (SRs), especialmente com o crescimento de Redes Sociais Baseadas em Localização, como Foursquare, Gowalla e Yelp. A qualidade dessas recomendações é essencial para enriquecer a experiência do usuário nessas plataformas, facilitando a sociabilidade e promovendo o turismo, além de levantar uma série de desafios para a comunidade. No entanto, os sistemas tradicionais de recomendação de POIs frequentemente se limitam a considerar informações como avaliações de locais, fotos, horários de acesso e check-ins, negligenciando dados geográficos relevantes, como feições geográficas que incluem rios, edifícios, ruas e lagos no contexto de um POI. Essas feições podem influenciar significativamente as preferências dos usuários, uma vez que eles podem visitar um POI por gostar das feições geográficas presentes no ambiente. Como exemplo, algumas pessoas preferem cafeterias próximas a lagos e áreas arborizadas em vez de rodovias movimentadas. Neste estudo, propomos e avaliamos a utilização de embeddings geográficos que incorporam feições geográficas para aprimorar SRs de POIs. Os resultados indicaram que o uso dos embeddings que consideram as feições aumentou a acurácia e o MRR em até 1.65% na tarefa de recomendação do próximo POI no conjunto de dados utilizado, em comparação ao baseline, confirmando a importância das feições geográficas para melhorar SRs de POIs.

EVALUATION OF GEOGRAPHIC EMBEDDINGS IN POI RECOMMENDATION SYSTEMS

ABSTRACT

Points of Interest (POIs) recommendation gains prominence in the context of recommendation systems (RSs), especially with the growth of Location-Based Social Networks, such as Foursquare, Gowalla, and Yelp. The quality of these recommendations is essential to enrich user experience on these platforms, facilitating sociability and promoting tourism, in addition to raising a series of challenges for the community. However, traditional POI recommendation systems often limit themselves to considering information such as site reviews, photos, access times, and check-ins, neglecting relevant geographical data, such as geographical features that include rivers, buildings, streets, and lakes in POIs' vicinity. These geographic features can significantly influence user preferences, as they may visit a POI because they like the geographical features present in the environment. For example, some people prefer coffee shops near lakes and wooded areas instead of busy highways. In this study, we propose and evaluate the use of geographical embeddings that incorporate geographical features to enhance POI RSs. The results indicated that the use of embeddings that consider the features increased the accuracy and the MRR by up to 1.65% in the task of next POI recommendation in the used dataset, compared to the baseline, confirming the importance of geographical features to improve POI RSs.

Avaliação de embeddings geográficos em Sistemas de Recomendação de POIs

Trabalho de Conclusão de Curso

Nícolas Moreira Nobre Leite (Aluno), Cláudio Campelo (Orientador)

Departamento de Sistemas e Computação

Universidade Federal de Campina Grande

Campina Grande, Paraíba - Brasil

RESUMO

A recomendação de Pontos de Interesse (POIs) ganha destaque no contexto de sistemas de recomendação (SRs), especialmente com o crescimento de Redes Sociais Baseadas em Localização, como Foursquare, Gowalla e Yelp. A qualidade dessas recomendações é essencial para enriquecer a experiência do usuário nessas plataformas, facilitando a sociabilidade e promovendo o turismo, além de levantar uma série de desafios para a comunidade. No entanto, os sistemas tradicionais de recomendação de POIs frequentemente se limitam a considerar informações como avaliações de locais, fotos, horários de acesso e *check-ins*, negligenciando dados geográficos relevantes, como feições geográficas que incluem rios, edifícios, ruas e lagos no contexto de um POI. Essas feições podem influenciar significativamente as preferências dos usuários, uma vez que eles podem visitar um POI por gostar das feições geográficas presentes no ambiente. Como exemplo, algumas pessoas preferem cafeterias próximas a lagos e áreas arborizadas em vez de rodovias movimentadas. Neste estudo, propomos e avaliamos a utilização de *embeddings* geográficos que incorporam feições geográficas para aprimorar SRs de POIs. Os resultados indicaram que o uso dos *embeddings* que consideram as feições aumentou a acurácia e o *MRR* em até 1.65% na tarefa de recomendação do próximo POI no conjunto de dados utilizado, em comparação ao *baseline*, confirmando a importância das feições geográficas para melhorar SRs de POIs.

PALAVRAS-CHAVE

Sistemas de recomendação, Pontos de Interesse, *embeddings* geográficos, feições geográficas, Aprendizagem de Máquina

1 INTRODUÇÃO

Atualmente, os sistemas de recomendação (SRs) desempenham um papel crucial na personalização da experiência do usuário, fornecendo sugestões altamente relevantes em uma variedade de domínios, incluindo *e-commerces* [1], entretenimento [13] e música [21]. Esses sistemas são fundamentais para ajudar os usuários a navegar por diversas opções, recomendando produtos, filmes e músicas que se alinham com suas preferências.

Dentro do vasto campo de SRs, a recomendação de Pontos de Interesse (POIs) representa uma subárea de crescente interesse.

Os autores retêm os direitos, ao abrigo de uma licença Creative Commons Atribuição CC BY, sobre todo o conteúdo deste artigo (incluindo todos os elementos que possam conter, tais como figuras, desenhos, tabelas), bem como sobre todos os materiais produzidos pelos autores que estejam relacionados ao trabalho relatado e que estejam referenciados no artigo (tais como códigos fonte e bases de dados). Essa licença permite que outros distribuam, adaptem e evoluam seu trabalho, mesmo comercialmente, desde que os autores sejam creditados pela criação original.

POIs, que podem ser locais como restaurantes, hotéis e parques, são essenciais para experiências de turismo, um setor que contribui significativamente para o crescimento econômico global [4]. No entanto, a recomendação eficaz de POIs apresenta desafios únicos, devido à complexidade inerente em capturar as preferências dos usuários, que são influenciadas por fatores culturais, pessoais e geográficos.

Sistemas tradicionais de recomendação de POIs frequentemente se baseiam em *embeddings* que consideram sempre as mesmas informações, como avaliações dos usuários sobre os locais, *check-ins*, horários de visitação e tipos de POIs [15, 32, 35]. Esses dados são bastante relevantes pois representam padrões de visitação e características dos POIs visitados. Os tipos de POI, por exemplo, categorizam os pontos de interesse de diversas formas, como comerciais (por exemplo, restaurantes, hotéis), recreativos (parques, museus) e transportes (aeroportos, estações ferroviárias), facilitando buscas em bancos de dados geográficos e agrupamentos. Entretanto, apesar das recentes pesquisas mostrarem bons resultados [31–33], elas não consideram a utilização de feições geográficas no contexto de um POI. Essas feições podem ser cruciais para caracterizar as preferências dos usuários, podendo inclusive motivar a escolha dos POIs a serem visitados. Um exemplo disso é que algumas pessoas podem ir a um parque com um lago, como o Central Park em Nova York, e em seguida para um restaurante com as mesmas características, com um lago ou rio próximos.

Diante disso, conduzimos um estudo com o objetivo de responder à seguinte questão: a utilização de *embeddings* geográficos que incorporam feições geográficas pode aprimorar as recomendações de POIs? Nesse contexto, propomos uma abordagem para incorporar *embeddings* dessa natureza em um modelo de recomendação de POIs e avaliamos experimentalmente nossa solução a partir da comparação da eficácia do recomendador antes e depois dessa adaptação.

Para isso, selecionamos uma abordagem recentemente desenvolvida para incorporar feições geográficas em *embeddings* de tipos de POIs (GeoContext2Vec [23]), que tem se mostrado promissora para outros cenários de aplicação. O modelo selecionado para implementação da abordagem foi o GETNext [32], referenciado na literatura como uma das principais opções para recomendação de POI. Os resultados obtidos indicam que a abordagem parece promissora, conseguindo capturar a importância das feições geográficas presentes no contexto de um POI para a escolha do próximo POI a ser visitado por um usuário.

As seções subsequentes estão organizadas da seguinte maneira: a Seção 2 explica conceitos relevantes para esta pesquisa. A Seção 3 descreve trabalhos relacionados na área de representação de POIs e

SRs de POIs. Na Seção 4 apresentamos nossa metodologia, incluindo a arquitetura do modelo utilizado e geração dos *embeddings* geográficos. A Seção 5 discute os resultados de nossos experimentos. Por fim, a Seção 6 conclui o artigo e aponta para pesquisas futuras.

2 FUNDAMENTAÇÃO TEÓRICA

Esta seção aborda conceitos teóricos relevantes no contexto de aprendizado de máquina e SRs de POIs abordados nessa pesquisa, como *Word Embeddings*, *Transformers* e feições geográficas.

2.1 Word Embeddings

*Embeddings*¹ são representações vetoriais densas e de baixa dimensão de objetos em um espaço contínuo, que podem incluir palavras, frases, documentos, imagens e sons. A principal função dos *embeddings* é converter dados de alta dimensionalidade e frequentemente esparsos em vetores mais compactos e densos, preservando relações semânticas e contextuais, facilitando a manipulação matemática eficiente.

Word Embeddings são essencialmente uma categoria específica de *embeddings* focados exclusivamente em palavras, permitindo que modelos de aprendizado de máquina e processamento de linguagem natural (PLN) capturem aspectos semânticos, sintáticos e relações de contexto entre palavras. A principal função dos *Word Embeddings* é transformar palavras, que são símbolos discretos e sem uma representação numérica natural, em vetores contínuos que podem ser processados por algoritmos de aprendizado de máquina.

Os *Word Embeddings* são amplamente utilizados em tarefas como análise de sentimentos [8], reconhecimento de entidades nomeadas [9], tradução automática [38], entre outros. Eles são fundamentais porque permitem que o modelo entenda que palavras com significados ou contextos semelhantes tenham representações vetoriais próximas. Na Figura 1 podemos observar as palavras "gato", "cachorro" e "computador" representadas como *embeddings* em um espaço vetorial. Nessa figura, é possível perceber que as palavras "gato" e "cachorro" estão mais próximas no espaço vetorial, ao contrário de "gato" e "computador". Isso demonstra a habilidade dos *embeddings* de capturar o significado semântico das palavras, fazendo com que vetores de palavras semelhantes estejam mais próximos entre si no espaço vetorial.

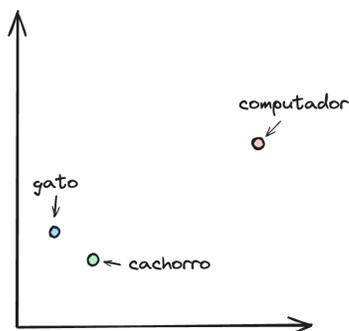


Figura 1: Representação visual de *embeddings* das palavras 'gato', 'cachorro' e 'computador'

¹<https://www.cloudflare.com/pt-br/learning/ai/what-are-embeddings/>

Um dos exemplos mais notáveis e amplamente utilizados de *Word Embeddings* é o Word2Vec, desenvolvido por Mikolov *et al.* [18] em 2013. O Word2Vec utiliza redes neurais para aprender representações vetoriais de palavras a partir de grandes conjuntos de dados textuais. Existem duas arquiteturas principais dentro do Word2Vec:

- **CBOV (Continuous Bag of Words):** Neste modelo, o objetivo é prever uma palavra com base no contexto. Por exemplo, dada a sequência de palavras "o gato dorme no", o modelo CBOV tentaria prever a palavra "sofá" como a palavra mais provável para completar a frase.
- **Skip-gram:** O modelo Skip-gram inverte a tarefa do CBOV, tentando prever o contexto a partir de uma palavra. Por exemplo, dada a palavra "gato", o modelo tentaria prever palavras como "dorme", "no" e "sofá".

Ambos os modelos são eficazes para capturar relações semânticas e sintáticas, podendo inclusive ser aplicado em situações que não envolvem palavras diretamente, como em sistemas de recomendação [2].

2.2 Transformers

Transformers é uma arquitetura de aprendizado de máquina que ganhou destaque no campo de PLN desde sua introdução em 2017 por Vaswani *et al.* [24]. Essa arquitetura baseia-se em mecanismos de atenção, especificamente a "atenção auto-dirigida" (*self-attention*). Esse mecanismo permite que o modelo avalie diferentes partes de uma entrada para determinar quais são mais relevantes entre si, aprendendo contextos e dependências complexas em dados sequenciais, independentemente da distância entre os elementos relevantes na sequência. Isso elimina a necessidade de redes neurais recorrentes (RNNs) ou convolucionais (CNNs), que eram comumente utilizadas em modelos anteriores para processamento de sequências. Por esse motivo, modelos que utilizam a arquitetura *Transformers* são excelentes no tratamento de longas dependências entre elementos de sequência de entrada, permitindo ainda processamento em paralelo.

A arquitetura dos *Transformers* é geralmente dividida em duas partes principais: o codificador (*encoder*) e o decodificador (*decoder*) (Figura 2). Cada parte consiste em uma pilha de camadas idênticas que contém subcamadas de atenção multi-cabeça e redes neurais *feed-forward* densamente conectadas. Normalizações de camada e conexões residuais também são empregadas para facilitar o treinamento de redes muito profundas.

Durante a fase de treinamento de um modelo que utiliza a arquitetura *Transformers*, o processo central envolve o ajuste dos pesos de atenção. Esses pesos são essenciais porque determinam quão importante é cada palavra da sequência de entrada em relação às outras. O modelo realiza isso calculando os pesos de atenção que ajudam a destacar partes específicas da entrada que são mais relevantes para a tarefa em questão.

A arquitetura *Transformers* é utilizada hoje em uma variedade de tarefas no campo de aprendizado de máquina, desde visão computacional, processamento de áudio, geração de texto, entre outros.

2.3 Feições geográficas

Feições geográficas referem-se a qualquer parte da superfície terrestre ou a qualquer elemento presente nela que possa ser representado

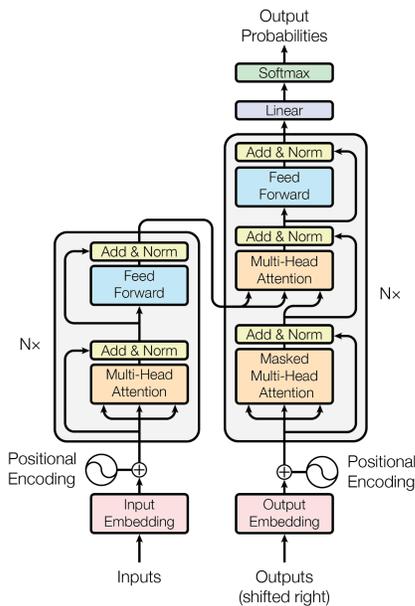


Figura 2: Arquitetura do *transformer*
Source: [24]

em um mapa². A identificação e análise dessas feições são cruciais em áreas como geografia e ciências ambientais, contribuindo para pesquisas em planejamento urbano, estudos ambientais, navegação e cartografia. Cada feição geralmente possui atributos associados que fornecem informações adicionais, como nome, população ou elevação, e podem apresentar características espaciais adicionais, como coordenadas geográficas. As feições podem representar diversos tipos de entidades geográficas, como pontos (Figura 3.a), linhas (Figura 3.b), polígonos (Figura 3.c) ou até objetos mais complexos, como redes ou superfícies. Cada um desses tipos pode representar informações diferentes dentro do contexto de uma cidade, com pontos³ representando uma feição que não tem comprimento nem área em uma determinada escala, como semáforos, árvores e fontes; linhas⁴, que são feições com comprimento mas sem área, como rios, ruas e rodovias; e polígonos⁵, que delimitam uma área em uma determinada escala, podendo representar edifícios, parques e lagos. Essas definições não são necessariamente excludentes; um rio, por exemplo, pode ser tratado tanto como uma feição do tipo linha quanto do tipo polígono.

3 TRABALHOS RELACIONADOS

Esta seção descreve trabalhos correlatos nas áreas de representação e SRs de POIs.

²<https://support.esri.com/pt-br/gis-dictionary/search?q=feição>

³<https://support.esri.com/pt-br/gis-dictionary/point-feature>

⁴<https://support.esri.com/pt-br/gis-dictionary/line-feature>

⁵<https://support.esri.com/pt-br/gis-dictionary/polygon-feature>

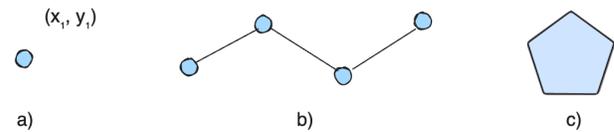


Figura 3: Tipos de feições geográficas: a) ponto b) linha c) polígono

3.1 Representação de POIs

Diversos estudos têm explorado a representação de POIs por meio de técnicas de *embedding*, destacando-se como uma abordagem promissora para melhorar a eficácia dos SRs de POIs. Estudos como o de Wang *et al.* [27], que introduziram o Urban2Vec, exemplificam inovações significativas nesse campo. O Urban2Vec é uma técnica não supervisionada que gera representações de bairros e vizinhanças utilizando uma combinação de imagens de *street view* e informações textuais dos POIs. Esta abordagem é notável porque integra características visuais e textuais com dados geoespaciais, superando os *baselines* existentes e competindo com métodos totalmente supervisionados em tarefas de predição *downstream* como planejamento urbano, desenvolvimento de modelos de negócios e melhoria do bem-estar social.

No contexto de SRs de POIs, os *embeddings* gerados por essas técnicas são utilizados para capturar a essência dos POIs e contextualizar as recomendações de acordo com as preferências dos usuários e características locais. Por exemplo, Zheng *et al.* [5] introduziram o modelo POI2Vec, que utiliza *embeddings* para representar os POIs com base nas interações dos usuários e nos *check-ins*, permitindo um entendimento mais profundo dos padrões de movimento e preferências.

Por outro lado, Silva *et al.* [23] propuseram o GeoContext2Vec, uma abordagem que enfatiza a importância das feições geográficas na vizinhança de um POI para a geração de *embeddings*. Diferente de métodos que se apoiam em dados de *check-in* ou descrições textuais, o GeoContext2Vec avalia a proporção e a unicidade do espaço ocupado pelas feições. Este método demonstrou superar o ITDL (Yan *et al.* [30]), um método estado-da-arte baseado em coocorrências de POIs, em termos de avaliação de similaridade de tipos de POI por análises humanas. Além disso, a utilização de mapas de domínio público como o OpenStreetMap (OSM) para a criação de vetores é destacada como uma alternativa prática e reproduzível.

Apesar de muitos estudos sobre SRs de POIs utilizarem *embeddings* gerados com base em tipos de POI, *check-ins* e outros fatores, identificamos uma lacuna no uso de *embeddings* que integram feições geográficas para aprimorar esses sistemas. Assim, empregamos o GeoContext2Vec para gerar *embeddings* de tipos de POIs que incluem feições geográficas, utilizando-os no contexto de recomendação de POIs, presente nessa pesquisa, para aprimorar as recomendações.

3.2 Sistemas Convencionais de Recomendação de POIs

Pesquisas iniciais na área de SRs de POIs utilizaram cadeias de Markov e técnicas de fatoração de matrizes [3, 29, 37]. Entretanto, essas abordagens se tornaram limitadas para representar os padrões de visitação dos usuários quando comparadas a abordagens baseadas em aprendizagem profunda e *embeddings* [7].

Além disso, muitas abordagens modelam as relações entre POIs visitados e potenciais POIs incorporando e priorizando fatores temporais. Yuan *et al.* [34] identificaram uma lacuna significativa nos métodos existentes, que frequentemente negligenciavam a influência do tempo específico do dia nas preferências dos usuários. Eles propuseram um modelo de recomendação colaborativa que integra informações temporais, permitindo recomendações mais precisas para diferentes momentos do dia. Shi *et al.* [22] introduziram uma rede de memória atencional com *embeddings* baseados em correlação (AMN-CE) para recomendação de POI considerando o tempo, propondo um mecanismo de atenção temporal para ajustar a influência de diferentes horários nas preferências do usuário. Wang *et al.* [25] desenvolveram um modelo que incorpora preferências temporais de *check-in* dos usuários, projetando uma rede neural entre-grafos para controlar o fluxo de informação através de diferentes espaços semânticos, melhorando a precisão das recomendações ao considerar a relação entre horários de *check-in* e POIs.

Halder *et al.* [10] abordaram a recomendação do próximo POI considerando o tempo de fila que os usuários levam para entrar em um POI, um fator crítico que influencia o comportamento de mobilidade do usuário. Utilizando um modelo *Transformers*, o TLRM, os autores não só recomendam o próximo POI, mas também preveem o tempo de fila necessário para um usuário entrar em um POI.

Embora os fatores temporais sejam importantes, a influência geográfica dos POIs também é crucial para aprimorar as recomendações. Diversas técnicas de geração de *embeddings* geográficos têm sido aplicadas considerando o contexto de POIs, vizinhança e características, além de fatores temporais, para capturar as preferências dos usuários e melhorar as recomendações.

3.3 Sistemas de Recomendação de POIs que consideram dados geográficos

Luo *et al.* [17] propuseram o *Spatio-Temporal Attention Network* (STAN), um modelo que aborda a questão da escassez espacial e das relações temporais empregando uma arquitetura de atenção em duas camadas. Este modelo é hábil em facilitar interações entre locais não adjacentes e *check-ins* não consecutivos, capturando os efeitos espaço-temporais explícitos que influenciam o comportamento do usuário. Wang *et al.* [26] propõem um método de recomendação de POIs que aproveita influências sequenciais, de categoria e geográficas. Este método começa extraíndo vetores latentes de POIs e preferências do usuário a partir de sequências de *check-in* usando um modelo de *embedding* de palavras. Em seguida, aplica-se uma filtragem colaborativa para prever as preferências do usuário para diferentes POIs com base em seu comportamento.

Qin *et al.* [19] propuseram o *framework Disentangled dual-graph* (DisenPOI) para recomendação do próximo POI a ser visitado, que

aprimora a recomendação de POIs ao separar as influências sequenciais e geográficas, frequentemente confundidas em abordagens tradicionais. O *framework* utiliza dois grafos distintos: um para modelar a sequência de visitas do usuário e outro para representar as relações geográficas entre POIs. Por meio de aprendizado contrastivo, o *framework* extrai representações dissociadas dessas influências, permitindo uma compreensão mais clara das preferências do usuário e resultando em recomendações mais precisas e interpretáveis. Yang *et al.* [32] apresentaram o modelo *Graph Enhanced Transformer* (GETNext), que também usa grafos para a recomendação do próximo POI, aproveitando um mapa de fluxo de trajetória global e uma arquitetura baseada em *Transformers*. Para capturar padrões genéricos de movimento de um usuário entre POIs, os autores construíram um grafo que representa a trajetória dos usuários. Nesse grafo, os nós refletem POIs com atributos como localização geográfica, categoria do POI e frequência de *check-in*. Após isso, eles utilizaram uma Rede Neural Convolutiva de Grafos para aprender os *embeddings* de POIs que encapsulam essas transições globais e recomendar o próximo POI a ser visitado.

Yan *et al.* [31] desenvolveram o *Spatio-Temporal HyperGraph Convolutional Network* (STHGCN), um modelo para recomendação do próximo POI que emprega um hipergrafo para analisar informações de trajetórias detalhadas. O modelo aprende com trajetórias históricas e colaborativas, resolvendo o problema de *cold start* [14] e aprimorando recomendações para diferentes durações de trajetórias de usuários. Utilizando um *Transformers* com hipergrafo, o STHGCN integra estruturas de hipergrafo com dados espaço-temporais, mostrando-se superior a métodos anteriores em testes práticos. Yin *et al.* [33] desenvolveram o modelo *Sequence-based Neighbour search and Prediction Model* (SNPM) para recomendação do próximo POI, empregando técnicas de *embeddings* de grafos e métodos Eigenmap para analisar relações de POIs a partir de dados esparsos de *check-in*. O modelo inclui um Grafo de Vizinhos Dinâmico e uma Previsão de Dependência de Múltiplos Passos, considerando tanto o estado atual quanto sequências históricas de visitas aos POIs.

Embora as abordagens previamente citadas tenham avançado significativamente ao considerar dados geográficos e arquiteturas baseadas em grafos para melhorar as recomendações, ainda existe uma lacuna importante na integração de feições geográficas em modelos de recomendação de POIs. Sendo assim, propomos uma abordagem que utiliza *embeddings* geográficos de tipos de POI que incorporam feições geográficas, como descrito em [23]. Nós então integramos esses *embeddings* na arquitetura baseada em grafos e *Transformers* do modelo GETNext [32] e verificamos se a adição desses *embeddings* melhora as recomendações de POIs.

4 METODOLOGIA

Em nossa abordagem metodológica, empregamos a geração de *embeddings* de tipos de POIs utilizando o GeoContext2Vec e então integramos esses *embeddings* na arquitetura baseada em grafos e *Transformers* do modelo GETNext. Os detalhes destes processos serão explorados nas subseções seguintes.

4.1 Geração de *embeddings* de tipos de POI

Como mencionado anteriormente, as feições geográficas desempenham um papel fundamental na representação de tipos de POIs. SRs de POIs podem se beneficiar significativamente disso, pois tentam capturar as preferências dos usuários com base em diferentes atributos.

Para isso, utilizamos o GeoContext2Vec para representar os tipos de POIs, considerando seu potencial em aproveitar as feições geográficas presentes no contexto de um POI. Diferentemente dos métodos tradicionais que se baseiam em coocorrências, *check-ins* e popularidade de POIs, o GeoContext2Vec explora as informações geográficas disponíveis em mapas, como o OSM, para gerar *embeddings* dos tipos de POI de forma mais precisa. O OSM possui quatro tabelas principais que utilizamos para extrair as feições geográficas relevantes para o nosso contexto. São elas: *planet_osm_polygons*, que contém feições como edifícios, parques, rios, lagos, etc; *planet_osm_lines*, que inclui ruas e rodovias menores, ferrovias, etc; *planet_osm_roads*, que encapsula as principais estradas e rodovias da cidade; e *planet_osm_points*, que possui semáforos, árvores, fontes, etc.

Inicialmente, o algoritmo do GeoContext2Vec considera um raio ao redor do POI (por exemplo, 100 metros) e identifica as feições geográficas presentes nesse contexto (Figura 4). Em seguida, ele calcula a proporção do espaço ocupado por cada feição geográfica no contexto. Por exemplo, um rio que atravessa todo o contexto terá uma proporção maior do que um pequeno edifício.

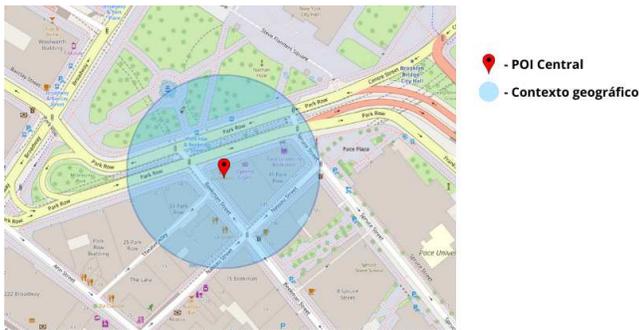


Figura 4: Contexto geográfico de um POI

Além disso, o algoritmo considera um parâmetro ω que determina a ênfase dada à área das feições geográficas em relação às suas ocorrências. Se ω for definido como 1.0, apenas a área das feições será considerada. Por outro lado, se ω for 0.0, apenas as ocorrências das feições serão levadas em conta. Valores entre 0.0 e 1.0 indicam uma combinação das duas, onde a proporção é determinada pelo valor do parâmetro. O algoritmo também avalia a unicidade de cada feição geográfica. Um único rio, por exemplo, será mais relevante para caracterizar o contexto do que pequenas árvores presentes em grande quantidade. Combinando a proporção de espaço e a unicidade, o algoritmo gera um fator que aumenta a relação de coocorrência entre o tipo de POI e cada feição geográfica presente no contexto. Por fim, o conjunto de relações de coocorrência gerado é usado para treinar um modelo Word2Vec [18], que aprende *embeddings* vetoriais para cada tipo de POI. Esses *embeddings* capturam as relações contextuais entre os tipos de POI e as feições geográficas do contexto.

O algoritmo foi aplicado para a geração de *embeddings* considerando os tipos de POIs presentes no *dataset* utilizado. Definimos um raio de 400m para o contexto geográfico de um POI e consideramos todas as feições geográficas presentes na circunferência desse raio. Esse valor foi estabelecido por meio de experimentação, sendo 400m o que produziu os resultados mais favoráveis. Também definimos o parâmetro ω como 0.8, o que significa que 80% é destinado à área da feição e 20% às ocorrências, que também se mostrou mais apropriado nas nossas experimentações. Após isso, extraímos todas as feições relevantes considerando as tabelas do OSM mencionadas anteriormente e em seguida calculamos o fator de coocorrência para cada tipo de POI e feição geográfica. Esse fator considera tanto a proporção do espaço ocupado pela feição quanto sua unicidade no contexto, e tenta capturar a relevância de uma feição em relação a um tipo específico de POI dentro de um dado contexto. Com isso, formamos um conjunto de treinamento onde cada entrada reflete uma relação de coocorrência, ponderada pelo fator calculado. Treinamos então um modelo Word2Vec usando a arquitetura *Skip-Gram*, que aprende representações vetoriais para cada tipo de POI com base nas feições geográficas associadas, e obtemos um *embedding* para cada tipo de POI presente no *dataset*, refletindo as feições geográficas de seu contexto.

4.2 Arquitetura do Modelo

O modelo GETNext tenta prever o próximo POI a ser visitado por um usuário, integrando contexto e informações sequenciais. A arquitetura empregada no modelo inclui componentes cruciais para essa previsão, mesclando padrões de movimento genéricos, preferências específicas do usuário e dados contextuais para prever o próximo POI a ser visitado. A utilização de redes neurais de grafos, mecanismos de atenção e a arquitetura baseada em *Transformers* possibilita a captura de relações complexas e recomendações personalizadas.

Os três principais componentes da arquitetura são: Aprendizado de Movimento Genérico, *Embedding* Contextual e Codificador-Decodificador. Esses três componentes são descritos nas subseções seguintes.

4.2.1 Aprendizado de Movimento Genérico (Generic Movement Learning). Na camada de Aprendizado de Movimento Genérico, uma Rede Neural Convolutiva de Grafos (GCN) analisa dados históricos de *check-in* para aprender os *embeddings* de POIs. Esses *embeddings* refletem os padrões de movimento dos usuários entre POIs, considerando aspectos como localização, categoria e frequência de visitas. Adicionalmente, a camada inclui um Mapa de Atenção de Transição, que modela explicitamente as probabilidades de transição entre POIs, reforçando a influência dos padrões coletivos de movimento na previsão.

4.2.2 Embedding Contextual (Contextual Embedding). Na camada de *Embedding* Contextual são gerados outros três *embeddings*, além dos *embeddings* de POIs gerados na camada de Aprendizado do Movimento Genérico. São eles: *embeddings* de usuário, categoria e tempo. Esses *embeddings* são projetados para capturar, respectivamente, as preferências do usuário, a classificação dos POIs e as preferências temporais. No entanto, para extrair compreensões

mais profundas, é necessário combinar esses *embeddings*. São integrados então os *embeddings* de POI com os específicos do usuário, que são derivados do histórico individual, a fim de personalizar as recomendações. Além disso, os *embeddings* temporais e os *embeddings* de categoria também são combinados, reconhecendo que as preferências dos usuários podem mudar conforme a categoria do POI e os diferentes horários do dia. Essas integrações ajustam o modelo para alinhar-se às preferências do usuário e ao contexto temporal das visitas aos POIs.

4.2.3 Codificador-Decodificador. A camada de codificador-decodificador, subdividida em um codificador *Transformer* e decodificadores *perceptron* multi-camada, processa sequências de *check-ins* e extrai características relevantes usando camadas de atenção e redes totalmente conectadas. O decodificador emprega um *perceptron* multi-camadas para fazer previsões detalhadas, incluindo o próximo POI a ser visitado, sua categoria e o tempo de visita.

Dado o contexto da arquitetura descrita, adicionamos os *embeddings* dos tipos de POI que incorporam as feições geográficas, gerados para a cidade de Nova York utilizando o *GeoContext2Vec*, a fim de que o modelo seja capaz de identificar os padrões de visita e associá-los às feições geográficas. Os *embeddings* foram adicionados após a concatenação dos *embeddings* combinados (POI + usuário e categoria + tempo), como mostrado na camada de *Embedding* Contextual na Figura 5. Essa adição tem como objetivo ter uma representação final que apresente as duas configurações, visita e feições. Essa abordagem pode ser intuitivamente compreendida como um meio de fornecer ao modelo uma compreensão mais rica do contexto geográfico associado aos locais visitados. Ao fazer isso, o modelo pode aprender a associar aspectos geográficos específicos com padrões de visita, potencialmente melhorando sua capacidade de fazer recomendações precisas e contextualmente relevantes.

5 EXPERIMENTOS E RESULTADOS

Esta seção descreve o *dataset*, métricas de avaliação e experimentos realizados para a tarefa de recomendação do próximo POI na cidade de Nova York.

5.1 Dataset

Conduzimos nossos experimentos no *dataset* público de Nova York do Foursquare¹, que possui 1.075 usuários, 5.099 POIs, 318 categorias e 104.074 *check-ins* coletados entre abril de 2012 e fevereiro de 2013. A sequência total de *check-ins* de um usuário foi dividida em trajetórias de intervalos de 24 horas, gerando um total de 14.160 trajetórias. Trajetórias que continham apenas um *check-in* foram eliminadas do *dataset*. O *dataset* foi dividido em treino, validação e teste seguindo uma ordem cronológica, sendo 80% para treino, 10% para validação e 10% para teste. Um ponto importante é que, se um usuário ou POI não apareceu no treinamento mas apareceu no teste, nós o ignoramos ao calcular as métricas.

5.2 Avaliação

Como métricas principais, utilizamos a acurácia@ k ($Acc@k$) e o MRR , que são usuais em SRs. A $Acc@k$ verifica se o POI que o

usuário visitou (ou interagiu) está presente entre os K principais POIs recomendados. Entretanto, como a $Acc@k$ não leva em conta a ordem dos POIs relevantes nos primeiros k resultados, utilizamos também o MRR , que considera a posição do primeiro POI relevante na lista de recomendações. Um MRR mais alto indica que, em média, o primeiro POI relevante aparece em uma posição superior nas listas de recomendações, refletindo em uma melhor qualidade. A $Acc@k$ foi definida pela seguinte equação:

$$Acc@k = \frac{\text{Qtd itens relevantes nos top } k \text{ recomendados}}{k} \quad (1)$$

Além disso, o MRR foi definido como:

$$MRR = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{\text{rank}_i} \quad (2)$$

onde Q representa um conjunto de consultas e rank_i é a posição do primeiro item relevante na i -ésima consulta. Por exemplo, se o sistema recomenda os locais na ordem B, A, D, C, E e o primeiro local relevante para o usuário é A, na segunda posição, o rank do primeiro item relevante rank_i é 2. Se essa fosse a única consulta considerada, o MRR seria calculado como:

$$MRR = \frac{1}{1} \sum_{i=1}^1 \frac{1}{2} = 0.5$$

Se houvesse mais consultas, somaríamos os recíprocos das posições dos primeiros itens relevantes de cada consulta e dividiríamos pelo número total de consultas para obter o MRR . Em ambos os casos, tanto para a $Acc@k$ quanto para o MRR , quanto maiores os valores, melhor.

5.3 Configuração do Experimento

Utilizamos a biblioteca do PyTorch para o treinamento dos modelos (GETNext e Word2Vec) e realização dos testes, em uma configuração de hardware que inclui um processador Intel Core i7-12700F 12 geração, 64GB de memória RAM e uma GPU NVIDIA GeForce RTX 4090 24GB. Os ajustes principais dos hiperparâmetros para o GETNext incluem a utilização de *embeddings* de 128 dimensões para POI e usuário, e 32 para tempo e categoria de POI, escolhidos com base nos mesmos valores definidos no trabalho de referência [32]. Além disso, utilizamos os *embeddings* gerados pelo *GeoContext2Vec* que possuem 70 dimensões. Por fim, o treinamento do modelo GETNext com a adição dos *embeddings* de tipos de POI que incorporam as feições geográficas foi realizado em 200 épocas e *batch* de tamanho 20. O código do fonte utilizado nessa pesquisa está disponível em um repositório do Github⁶.

5.4 Resultados

Nossa pesquisa focou na integração de *embeddings* de tipos de POIs, que incorporam feições geográficas, a um modelo de recomendação de POIs [32] com o objetivo de aprimorar as recomendações. Os resultados, mostrados na Tabela 1, indicam um aumento de até 1.65% nas métricas calculadas para o *dataset* de Nova York, em comparação com o GETNext, nosso principal *baseline*, e outros *baselines* comumente utilizados na área de SRs de POIs [6, 11, 12, 16,

¹<https://sites.google.com/site/yangdingqi/home/foursquare-dataset>

⁶<https://github.com/nicolasmnl/poi-recommendation-TCC>

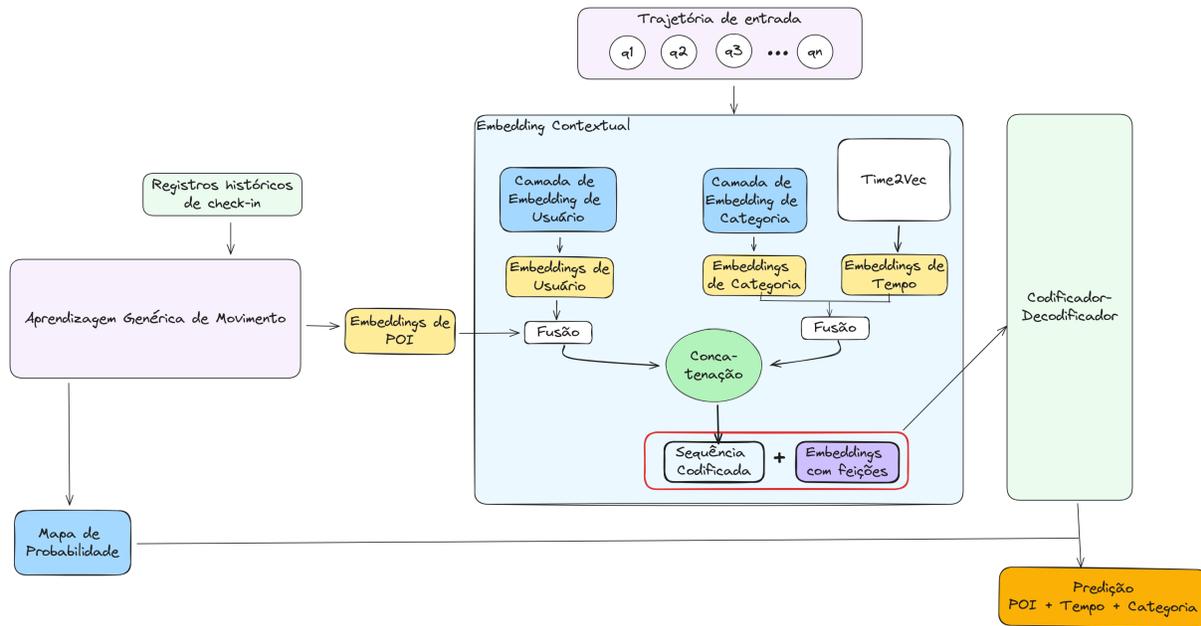


Figura 5: Arquitetura do modelo baseada no GETNext com *embeddings* dos tipos de POI que incluem feições geográficas

17, 20, 28, 36]. Os resultados do GETNext, apresentados na Tabela 1, foram obtidos reproduzindo os experimentos nas mesmas condições dos realizados pelos autores originais.

Tabela 1: Comparação de desempenho em Acurácia@K e MRR no *dataset* de Nova York

	Acc@1	Acc@5	Acc@10	Acc@20	MRR
MF [12]	0.0368	0.0961	0.1522	0.2375	0.0672
FPMC [20]	0.1003	0.2126	0.2970	0.3323	0.1701
LSTM [11]	0.1305	0.2719	0.3283	0.3568	0.1857
PRME [6]	0.1159	0.2236	0.3105	0.3643	0.1712
ST-RNN [16]	0.1483	0.2923	0.3622	0.4502	0.2198
STGCN [36]	0.1799	0.3425	0.4279	0.5214	0.2788
PLSPL [29]	0.1917	0.3678	0.4523	0.5370	0.2806
STAN [17]	0.2231	0.4582	0.5734	0.6328	0.3253
GETNext [32]	0.2268	0.4897	0.5880	0.6545	0.3460
GETNext com feições	0.2329	0.4987	0.6045	0.6670	0.3524
Aumento (%)	0.61%	0.9%	1.65%	1.25%	0.64%

Os resultados sugerem que a utilização de feições geográficas é promissora, pois reflete o comportamento real dos usuários, que possivelmente consideram o contexto geográfico de um POI ao decidir o próximo POI a visitar. Esses resultados também respondem à questão de pesquisa levantada, uma vez que *embeddings* que incorporam feições geográficas podem sim aprimorar as recomendações.

Além disso, nossa estratégia pode ser facilmente replicada, sendo necessário apenas adicionar os *embeddings* de tipos de POIs que

incorporam feições geográficas em outras abordagens com características semelhantes. Por exemplo, em [31], os autores desenvolveram um modelo que emprega características semelhantes às desta pesquisa, também superando o GETNext, mas se diferencia por não usar as feições geográficas e por utilizar um hipergrafo para analisar tanto as trajetórias históricas individuais dos usuários (intra-usuário) quanto as trajetórias colaborativas entre diferentes usuários (inter-usuário). Este modelo poderia ser aprimorado com a adição dos *embeddings* que consideram as feições geográficas, potencialmente melhorando os resultados apresentados. No entanto, a reprodução do trabalho [31] com a adição dos *embeddings* que incorporam as feições não está no escopo desta pesquisa.

6 CONCLUSÕES E TRABALHOS FUTUROS

SRs de POIs são fundamentais para enriquecer a experiência do usuário, fomentar a sociabilidade e impulsionar o turismo em várias regiões. Entretanto, sistemas convencionais de recomendação de POIs frequentemente se limitam a usar informações como avaliações de usuários, *check-ins*, horários de visitação e tipos de POIs, desconsiderando dados geográficos relevantes, como as feições geográficas no contexto dos POIs. Neste estudo, propomos a utilização de *embeddings* de tipos de POI que integram feições geográficas para melhorar os SRs de POIs. Avaliamos nossa abordagem principalmente em relação ao GETNext, além de outros *baselines* usuais em SRs de POIs. Os resultados mostram que a inclusão de feições geográficas nos *embeddings* de tipos de POI elevou as métricas em até 1.65% em relação ao GETNext no *dataset* de Nova York utilizado, demonstrando a relevância das feições geográficas na escolha dos usuários pelo próximo POI a visitar e respondendo à questão de pesquisa levantada.

Como limitações de nossos experimentos, notamos que em grandes metrópoles como Nova York, as feições geográficas presentes no contexto dos POIs são muito similares ao longo de toda a cidade, o que pode dificultar a distinção e associação entre um POI e seu contexto geográfico. Além disso, é importante mencionar que o algoritmo foi testado em apenas um *dataset*, referente ao de Nova York retirado do Foursquare, o que pode sugerir limitações na generalização dos resultados.

Para trabalhos futuros, pretendemos realizar experimentos com uma variedade maior de *datasets*, que incluam diferentes cidades e contextos geográficos, para avaliar a robustez e a capacidade de generalização do modelo proposto. Isso proporcionará uma compreensão mais profunda de como as feições geográficas afetam as recomendações de POIs em diversos ambientes. Adicionalmente, queremos explorar a integração dos *embeddings* de tipos de POI utilizados nesta pesquisa em modelos estado da arte de recomendação de POIs. O objetivo é verificar se a inclusão de feições geográficas pode potencializar o desempenho desses modelos também. Por fim, estudos específicos sobre quais feições geográficas contribuem mais para a melhoria das recomendações podem ser realizados, envolvendo análises de sensibilidade ou avaliações da importância relativa de diferentes tipos de feições em variados contextos, fornecendo informações valiosas para o aprimoramento contínuo dos SRs de POIs.

AGRADECIMENTOS

Gostaria de expressar minha gratidão às pessoas que me apoiaram não apenas durante esta pesquisa, mas também ao longo de toda a minha jornada no curso de Ciência da Computação na UFCG.

Em primeiro lugar, à minha família: meus pais, Adrio e Ana, e meus irmãos, Naími e Niago. Vocês nunca duvidaram do meu potencial e estiveram ao meu lado nos momentos bons e ruins. Agradeço pelos conselhos, orientações, motivações e apoio em todos os momentos. Vocês me fizeram refletir não apenas sobre minha vida acadêmica e profissional, mas principalmente sobre minha vida pessoal, me fazendo a pessoa que sou hoje (e até que gosto dela)! Posso afirmar com sinceridade que tenho o melhor time titular em casa! Amo vocês profundamente! Gostaria de dedicar uma seção inteira deste trabalho a vocês, mas, infelizmente, estou sendo avaliado e isso torna o processo mais difícil.

À minha noiva e futura esposa, Andrielly: Eu te amo! Não consigo pensar em outra expressão para colocar aqui. Obrigado por compartilhar sorrisos e lágrimas comigo, e pelas discussões e debates que são intrínsecos às nossas personalidades (fazer o quê se é coisa de gênio!). Estou aqui para o que você precisar, sempre!

Ao meu orientador e professor, Cláudio Campelo, imagino que já esteja cansado de me ver. Concluí todas as suas disciplinas, trabalhei com o senhor em diversos projetos e ainda o tive como meu orientador, convite esse que o senhor aceitou prontamente. Agradeço pelos ensinamentos e pelo vasto aprendizado durante toda a minha graduação.

Por fim, mas não menos importante, gostaria de agradecer aos amigos que fiz durante a minha jornada. Sem vocês, acredito que essa jornada não teria valido nem um pouco a pena. Henrique, Helen, Davi, Andressa, Kennedy, Cilas, Filipe, Salatiel, Matias, Tuza, Bia, Mariane, Sheila e Arthur, obrigado a todos vocês pelas risadas,

aprendizados, ensinamentos e desafios que enfrentamos juntos. A vida é assim, e embora possamos pensar que a parte mais difícil já passou, ainda há muito a ser vivido! Amo vocês e agradeço por tudo!

REFERÊNCIAS

- [1] R. Bhuvanya and M. Kavitha. 2023. A real-time e-commerce accessories recommender system by coupling deep learning and histogram features. *J. Intell. Fuzzy Syst.* (2023). <https://doi.org/10.3233/JIFS-223754>
- [2] Hugo Caselles-Dupré, Florian Lesaint, and Jimena Royo-Letelier. 2018. Word2vec applied to recommendation: hyperparameters matter. In *Proceedings of the 12th ACM Conference on Recommender Systems (RecSys '18)*. Association for Computing Machinery, New York, NY, USA, 352–356. <https://doi.org/10.1145/3240323.3240377>
- [3] Mehri Davtalan and Ali Asghar Alesheikh. 2021. A POI recommendation approach integrating social spatio-temporal information into probabilistic matrix factorization. *Knowledge and Information Systems* 63, 1 (01 Jan 2021), 65–85. <https://doi.org/10.1007/s10115-020-01509-5>
- [4] Kizito Uyi Ehigiamusoe. 2020. Tourism, growth and environment: analysis of non-linear and moderating effects. *Journal of Sustainable Tourism* 28, 8 (2020), 1174–1192. <https://doi.org/10.1080/09669582.2020.1729164>
- [5] Shanshan Feng, Gao Cong, Bo An, and Yeow Meng Chee. 2017. POI2Vec: Geographical Latent Representation for Predicting Future Visitors. *Proceedings of the AAAI Conference on Artificial Intelligence* 31, 1 (Feb. 2017). <https://doi.org/10.1609/aaai.v31i1.10500>
- [6] Shanshan Feng, Xutao Li, Yifeng Zeng, Gao Cong, Yeow Meng Chee, and Quan Yuan. 2015. Personalized Ranking Metric Embedding for Next New POI Recommendation. In *Proceedings of the twenty-fourth international joint conference on artificial intelligence*, Qiang Yang and Michael J. Wooldridge (Eds.). Association for the Advancement of Artificial Intelligence (AAAI), United States, 2069–2075. The Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI ; Conference date: 25-07-2015 Through 31-07-2015.
- [7] Shanshan Feng, Lucas Vinh Tran, Gao Cong, Lisi Chen, Jing Li, and Fan Li. 2020. HME: A Hyperbolic Metric Embedding Approach for Next-POI Recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '20)*. Association for Computing Machinery, New York, NY, USA, 1429–1438. <https://doi.org/10.1145/3397271.3401049>
- [8] Maria Giatsoglou, Manolis G Vozalis, Konstantinos Diamantaras, Athena Vakali, George Sarigiannidis, and Konstantinos Ch Chatzivasvas. 2017. Sentiment analysis leveraging emotions and word embeddings. *Expert Systems with Applications* 69 (2017), 214–224.
- [9] Maryam Habibi, Leon Weber, Mariana Neves, David Luis Wiegandt, and Ulf Leser. 2017. Deep learning with word embeddings improves biomedical named entity recognition. *Bioinformatics* 33, 14 (07 2017), i37–i48. <https://doi.org/10.1093/bioinformatics/btx228> arXiv:https://academic.oup.com/bioinformatics/article-pdf/33/14/i37/50314882/bioinformatics_33_14_i37.pdf
- [10] Sajal Halder, Kwan Hui Lim, Jeffrey Chan, and Xiuzhen Zhang. 2021. Transformer-Based Multi-task Learning for Queuing Time Aware Next POI Recommendation. In *Advances in Knowledge Discovery and Data Mining*, Kamal Karlapalem, Hong Cheng, Naren Ramakrishnan, R. K. Agrawal, P. Krishna Reddy, Jaideep Srivastava, and Tanmoy Chakraborty (Eds.). Springer International Publishing, Cham, 510–523.
- [11] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation* 9, 8 (11 1997), 1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735> arXiv:<https://direct.mit.edu/neco/article-pdf/9/8/1735/813796/neco.1997.9.8.1735.pdf>
- [12] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix Factorization Techniques for Recommender Systems. *Computer* 42, 8 (2009), 30–37. <https://doi.org/10.1109/MC.2009.263>
- [13] Abhishek Kulkarni, Larry Powell, Shaina Murphy, Nanjie Rao, and Sharon Lynn Chu. 2023. *Everyday-Inspired Movies: Towards the Design of Movie Recommender Systems based on Everyday Life through Personal Social Media*. 160–169. https://doi.org/10.1007/978-3-031-42286-7_9
- [14] Blerina Lika, Kostas Kolomvatsos, and Stathes Hadjiefthymiades. 2014. Facing the cold start problem in recommender systems. *Expert Systems with Applications* 41, 4, Part 2 (2014), 2065–2073. <https://doi.org/10.1016/j.eswa.2013.09.005>
- [15] Baoping Liu, Yijun Su, Daren Zha, Neng Gao, and Ji Xiang. 2019. CARec: Content-Aware Point-of-Interest Recommendation via Adaptive Bayesian Personalized Ranking. *Aust. J. Intell. Inf. Process. Syst.* 15, 3 (2019), 61–68.
- [16] Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. 2016. Predicting the Next Location: A Recurrent Model with Spatial and Temporal Contexts. *Proceedings of the AAAI Conference on Artificial Intelligence* 30, 1 (Feb. 2016). <https://doi.org/10.1609/aaai.v30i1.9971>

- [17] Yingtao Luo, Qiang Liu, and Zhaocheng Liu. 2021. STAN: Spatio-Temporal Attention Network for Next Location Recommendation. In *Proceedings of the Web Conference 2021 (WWW '21)*. Association for Computing Machinery, New York, NY, USA, 2177–2185. <https://doi.org/10.1145/3442381.3449998>
- [18] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. *arXiv:cs.CL/1301.3781*
- [19] Yifang Qin, Yifan Wang, Fang Sun, Wei Ju, Xuyang Hou, Zhe Wang, Jia Cheng, Jun Lei, and Ming Zhang. 2023. DisenPOI: Disentangling Sequential and Geographical Influence for Point-of-Interest Recommendation. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining (WSDM '23)*. Association for Computing Machinery, New York, NY, USA, 508–516. <https://doi.org/10.1145/3539597.3570408>
- [20] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2010. Factorizing personalized Markov chains for next-basket recommendation. In *Proceedings of the 19th International Conference on World Wide Web (WWW '10)*. Association for Computing Machinery, New York, NY, USA, 811–820. <https://doi.org/10.1145/1772690.1772773>
- [21] Tairyu Saito and Eri Sato-Shimokawara. 2023. Music Recommender System Considering the Variations in Music Selection Criterion Using an Interactive Genetic Algorithm. In *Computer Information Systems and Industrial Management*, Khalid Saeed, Jiří Dvorský, Nobuyuki Nishiuchi, and Makoto Fukumoto (Eds.). Springer Nature Switzerland.
- [22] Meihui Shi, Derong Shen, Yue Kou, Tiezheng Nie, and Ge Yu. 2021. Attentional Memory Network with Correlation-based Embedding for time-aware POI recommendation. *Knowledge-Based Systems* 214 (2021), 106747. <https://doi.org/10.1016/j.knosys.2021.106747>
- [23] Salatiel Dantas Silva, Claudio Elízio Calazans Campelo, and Maxwell Guimarães De Oliveira. 2023. POI types characterization based on geographic feature embeddings. Association for Computing Machinery, New York, NY, USA. <https://doi.org/10.1145/3555776.3577659>
- [24] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2023. Attention Is All You Need. *arXiv:cs.CL/1706.03762*
- [25] Xin Wang, Xiao Liu, Li Li, Xiao Chen, Jin Liu, and Hao Wu. 2021. Time-aware User Modeling with Check-in Time Prediction for Next POI Recommendation. In *2021 IEEE International Conference on Web Services (ICWS)*. 125–134. <https://doi.org/10.1109/ICWS53863.2021.00028>
- [26] Xican Wang, Yanheng Liu, Xu Zhou, Xueying Wang, and Zhaoqi Leng. 2022. A Point-of-Interest Recommendation Method Exploiting Sequential, Category and Geographical Influence. *ISPRS International Journal of Geo-Information* 11, 2 (2022). <https://doi.org/10.3390/ijgi11020080>
- [27] Zhecheng Wang, Haoyuan Li, and Ram Rajagopal. 2020. Urban2Vec: Incorporating Street View Imagery and POIs for Multi-Modal Urban Neighborhood Embedding. *Proceedings of the AAAI Conference on Artificial Intelligence* 34, 01 (Apr. 2020), 1013–1020. <https://doi.org/10.1609/aaai.v34i01.5450>
- [28] Yuxia Wu, Ke Li, Guoshuai Zhao, and Xueming Qian. 2019. Long- and Short-term Preference Learning for Next POI Recommendation. 2301–2304. <https://doi.org/10.1145/3357384.3358171>
- [29] Yuxia Wu, Ke Li, Guoshuai Zhao, and Xueming Qian. 2022. Personalized Long- and Short-term Preference Learning for Next POI Recommendation. *IEEE Transactions on Knowledge and Data Engineering* 34, 4 (2022), 1944–1957. <https://doi.org/10.1109/TKDE.2020.3002531>
- [30] Bo Yan, Krzysztof Janowicz, Gengchen Mai, and Song Gao. 2017. From ITDL to Place2Vec: Reasoning About Place Type Similarity and Relatedness by Learning Embeddings From Augmented Spatial Contexts. In *Proceedings of the 25th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (SIGSPATIAL '17)*. Association for Computing Machinery, New York, NY, USA, Article 35, 10 pages. <https://doi.org/10.1145/3139958.3140054>
- [31] Xiaodong Yan, Tengwei Song, Yifeng Jiao, Jianshan He, Jiaotuan Wang, Ruopeng Li, and Wei Chu. 2023. Spatio-Temporal Hypergraph Learning for Next POI Recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '23)*. Association for Computing Machinery, New York, NY, USA, 403–412. <https://doi.org/10.1145/3539618.3591770>
- [32] Song Yang, Jiamou Liu, and Kaiqi Zhao. 2022. GETNext: Trajectory Flow Map Enhanced Transformer for Next POI Recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '22)*. ACM. <https://doi.org/10.1145/3477495.3531983>
- [33] Feiyu Yin, Yong Liu, Zhiqi Shen, Lisi Chen, Shuo Shang, and Peng Han. 2023. Next POI Recommendation with Dynamic Graph and Explicit Dependency. *Proceedings of the AAAI Conference on Artificial Intelligence* 37, 4 (Jun. 2023), 4827–4834. <https://doi.org/10.1609/aaai.v37i4.25608>
- [34] Quan Yuan, Gao Cong, Zongyang Ma, Aixin Sun, and Nadia Magnenat Thalmann. 2013. Time-aware point-of-interest recommendation. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '13)*. Association for Computing Machinery, New York, NY, USA, 363–372. <https://doi.org/10.1145/2484028.2484030>
- [35] Haoxiang Zhang, Wenchao Bai, Jingyi Ding, and Jiahui Jin. 2023. Time-Aware POI Recommendation Based on Multi-Grained Location Grouping. 1796–1801. <https://doi.org/10.1109/CSCWD57460.2023.10152604>
- [36] Pengpeng Zhao, Anjing Luo, Yanchi Liu, Jiajie Xu, Zhixu Li, Fuzhen Zhuang, Victor S. Sheng, and Xiaofang Zhou. 2022. Where to Go Next: A Spatio-Temporal Gated Network for Next POI Recommendation. *IEEE Transactions on Knowledge and Data Engineering* 34, 5 (2022), 2512–2524. <https://doi.org/10.1109/TKDE.2020.3007194>
- [37] Shenglin Zhao, Tong Zhao, Haiqin Yang, Michael Lyu, and Irwin King. 2016. STELLAR: Spatial-Temporal Latent Ranking for Successive Point-of-Interest Recommendation. *Proceedings of the AAAI Conference on Artificial Intelligence* 30, 1 (Feb. 2016). <https://doi.org/10.1609/aaai.v30i1.9986>
- [38] Will Y Zou, Richard Socher, Daniel Cer, and Christopher D Manning. 2013. Bilin-gual word embeddings for phrase-based machine translation. In *Proceedings of the 2013 conference on empirical methods in natural language processing*. 1393–1398.