Point-of-Interest Recommender Systems based on Location-Based Social Networks: A Survey from an Experimental Perspective

PABLO SÁNCHEZ*, Universidad Autónoma de Madrid, Spain
ALEJANDRO BELLOGÍN, Universidad Autónoma de Madrid, Spain

Point-of-Interest recommendation is an increasing research and developing area within the widely adopted technologies known as Recommender Systems. Among them, those that exploit information coming from Location-Based Social Networks (LBSNs) are very popular nowadays and could work with different information sources, which pose several challenges and research questions to the community as a whole. We present a systematic review focused on the research done in the last 10 years about this topic. We discuss and categorize the algorithms and evaluation methodologies used in these works and point out the opportunities and challenges that remain open in the field. More specifically, we report the leading recommendation techniques and information sources that have been exploited more often (such as the geographical signal and deep learning approaches) while we also alert about the lack of reproducibility in the field that may hinder real performance improvements.

1 INTRODUCTION

Recommender Systems (RSs) have risen as technological solutions to the information overload, as they help users to filter the most interesting items (in whatever domain the RS is being deployed) according to their preferences. Moreover, in the Internet era, they have become indispensable due to their ability to process large amounts of information and make personalized recommendations to users by learning their interests and tastes [102]. However, they serve other purposes as well. They are particularly useful to aggregate user behavior, which is pervasive nowadays, very common and easier to obtain thanks to the Internet and the increasing and diversity of social networks dealing with different domains. This is in fact related to the universal applicability of general RSs, since

---

*Both authors contributed equally to this research.

Authors' addresses: Pablo Sánchez, Information Retrieval Group, Universidad Autónoma de Madrid, Madrid, 28049, Spain, pablo.sanchezp@uam.es; Alejandro Bellogín, Information Retrieval Group, Universidad Autónoma de Madrid, Madrid, 28049, Spain, alejandro.bellogin@uam.es.

---

classic RSs have been oriented towards recommending music or movies, but for some years now they have been applied to other areas such as news, e-commerce, social contacts, healthcare, and tourism [4, 58, 59, 61, 78, 112].

In particular, Location-Based Social Networks (LBSNs) are a special kind of social Web systems where it is possible for users to register whenever they visit a specific Point-of-Interest (POI) through the so-called check-ins, or to establish social links with other users in the system [36]. In retrospective, they represent the digital versions of historical catalogs such as the Zagat survey or the Michelin guides, which aimed at summarizing and synthesizing ratings and reviews provided by amateur (since the beginning in Zagat around 1980s) or expert (since the 1930s in Michelin) food reviewers. These systems, as modern RSs based on LBSN data, had the same goal: reducing the choice overload of users, while providing a subjective measurement of the POI (for these two examples, restricted to restaurants) quality. Since then, location-based services that deliver information according to the location and context of the user and her device play a key role [56]. They appeared in the early 1990s, but thanks to the evolution of the technology (mobile devices, availability of GPS and navigation systems) a wide range of applications have emerged, not limited to LBSNs, but for gaming, health, fitness, and assistive technology. We refer the reader to [56] for a review on the research trends on that topic.

A popular demanded service in these LBSNs is POI recommendation. In general, these RS techniques aim at recommending users new places to visit when they arrive to a city or region; however, this problem is inherently multi-faceted and, hence, the following related problems are typically studied [15]: suggesting interesting previously unvisited places to a target user, recommending the next place to go, recommending events to attend and neighborhoods to explore in a urban setting, and discovering places in a city with respect to an input query and the user previous interests. Naturally, these recommendations are contextualized for a specific type of geographical region – such as a country, city, or town –, either implicitly (inferred from previous user history) [136] or explicitly (requested by the system itself) [104]. Additionally, a number of constraints could be incorporated into the model, such as type of trip (leasure or work), price, schedule, or weather [44, 112]. In this regard, commercial systems such as the early Triplehop's TripMatcher try to replicate the interactivity observed in typical sessions with travel agents [108], whereas recent platforms have created new services, such as exchange or sharing tourism-related products (Airbnb, Uber), integrating users in a community (TripAdvisor, Foursquare), searching and comparing (Trivago, Skyscanner), or booking and travel support (Expedia, Booking). We would also like to mention that, even though some of these companies have not yet fully exploited the potential of recommender systems – since they are more focused on tuning their filters according to the collected interactions [14] –, many researchers do make use of data from these companies to perform different types of recommendations. For example, it is well-known that hotel and/or tourist attraction recommendations can be performed using data from TripAdvisor or Booking [39, 106]). At the same time, another company from this domain, Expedia, organized in 2013 a contest[1] to find the best recommender for their website. However, the challenge was focused on data from search logs, so important characteristics from LBSNs (as we shall point out later) were ignored.

Finally, it is important to highlight that this domain exhibits other differences with respect to classical recommendation and how it has been modeled in the past, such as not being limited to a preference or rating matrix, or incorporating additional information such as geographical, social, or temporal signals to better adapt to the users' interests [81]. To obtain this type of information, researchers normally resort to exploiting LBSNs, as they include most of these attributes. It is worth mentioning that, even though the area of POI recommendation is of great interest to researchers

---

[1]Expedia contest: https://www.kaggle.com/c/expedia-personalized-sort

Table 1. Queries issued to the three digital libraries considered. For ScienceDirect the query is used in the field "Title, abstract or author-specified keywords", indicating 2011-2020 in the field "Years".

Source	Query
Scopus	((TITLE (point-of-interest) OR TITLE (venue) OR TITLE (poi) OR TITLE (location)) AND (TITLE (recommendation) OR TITLE (recommender)) AND (TITLE-ABS-KEY (lbsn) OR TITLE-ABS-KEY ("location-based social network")) AND (PUBYEAR > 2010)) AND (PUBYEAR < 2021) AND NOT TITLE (survey) AND (LIMIT-TO (LANGUAGE , "English"))
ScienceDirect	((lbsn) OR ("location-based social network")) AND -survey AND ("point of interest" OR venue OR location OR poi) AND (recommendation OR recommender)
ACM	[[Publication Title: "point-of-interest"] OR [Publication Title: "point of interest"] OR [Publication Title: poi] OR [Publication Title: venue] OR [Publication Title: location]] AND [[Publication Title: recommendation] OR [Publication Title: recommender]] AND [[Abstract: "location-based social network"] OR [Abstract: "lbsn"]] AND [Publication Date: (01/01/2011 TO 12/31/2020)]

because it allows to study the behavior and movement patterns of users, it is also appealing for companies and businesses in the tourism, leisure, and e-commerce domains, as they seek to attract and maintain customers by becoming popular and receiving good reviews – this is evidenced by the increasing number of companies working on related problems, as discussed before. As a consequence, a large number of articles have been published in recent years where different algorithms to recommend POIs were proposed by exploiting the information available in LBSNs. For that reason, we believe it is necessary to analyze the current proposals in the area for this type of recommendation, with special emphasis on the different types of implemented models and algorithms, information used, and evaluation methodologies followed in these works. This is because we consider all these pieces critical to produce real advances in current state-of-the-art, which might be hindered by reproducibility or evaluation issues [3, 103]. We present such an analysis in this systematic review, together with a careful and detailed discussion of the current problems that this area is facing at the moment, as well as potential future lines of research.

1.1 What are the differences between this survey and former ones?

Due to the growing interest in the general recommendation area on this domain, there is a considerable number of surveys related to POI recommendation and its different ramifications that complement our work. On the one hand, we have the works [8, 110, 133] which cannot be considered to be up to date anymore, since they were published 5 years ago, thus our survey should provide a novel overview of the works developed in this time. On the other hand, Gavalas et al. present in [44] an overview of optimization approaches that aim to solve a problem with applications on related tasks: the Tourist Trip Design Problem (TTDP); this can be applicable to route recommendation, which, as we specify in the next section, is not completely in the scope of this survey.

Additionally, we found some surveys that were too focused on specific subproblems. For instance, in [147] Zheng et al. consider the problem of location prediction but only based on Twitter information. Another example is [31], where Christoforidis et al. focus on deep learning techniques while neglecting the other types of recommendation algorithms.

Finally, since our analysis is also tailored towards the evaluation aspects of the works, it is worth mentioning those reviews where this aspect has been considered. However, we must acknowledge that we could not find any survey that focused on this particular aspect; because of that, we consider our survey is very valuable in this domain at the moment. To somehow overcome this shortcoming, we believe it is important to mention the experimental comparison presented by Liu et al. in [81], where they compared 12 recommendation models under different evaluation protocols and using

three datasets, which could help to analyze the behavior of those methods under the same and different conditions.

Table 2. Papers retrieved and final papers processed from the three digital libraries considered.

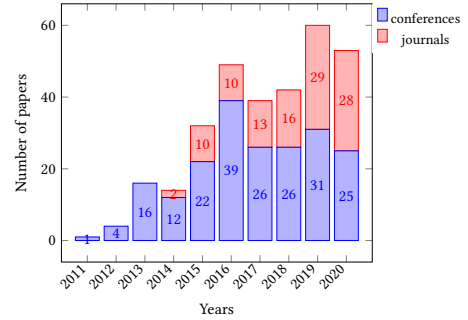| Source	Papers retrieved	Valid papers
Scopus	404	302
ScienceDirect	50	30
ACM	71	43
Unique papers	**431**	**310**

Fig. 1. Number of papers considered in this study based on their publication venue by year.

1.2 How do we collect the papers?

In order to select the papers that we have analyzed in this survey, we have searched in three digital libraries: Scopus[2], ScienceDirect[3], and ACM Digital Library[4]. As each library has a different query language to use within its search system, three different queries were needed to be defined and executed, however, they were designed to be as equivalent as possible[5].

The main characteristics these queries should satisfy are:

- Focus on articles between 2011 and 2020 (both included).
- Each publication should include in the title: "Point of interest recommendation" or (similar texts such as "POI recommender").
- Each publication should also include somewhere in the title, abstract, or keywords the terms "location-based social network" (or "LBSN"), since this survey is oriented to models using data coming from these systems, together with an analysis on the different evaluation methodologies that are being applied using datasets generated from LBSNs.

Thus, the final queries issued to each source are shown in Table 1. Based on this, Table 2 shows the number of papers we initially obtained with each query, as well as the actual number we finally analyzed. The difference was mostly caused to some papers not being available, some of them appeared in more than one source, and some had to be filtered out because they address a different task to the one we want to focus on this paper. We also decided to keep only those papers whose final goal is to recommend a list of POIs to each user; this includes related tasks such as next-POI recommendation as long as no trajectory or route recommendation is performed (as in [141]), but discards tasks such as route, category, or friend recommendation [24, 63, 111]. Hence, the only task we aim to cover with this review is POI recommendation (see later in Section 3.1 a formal definition of this problem, and in Section 3.5 other related tasks not covered herein).

Figure 1 shows the number of articles we include in this review according to their publication venue (conference or journal). We observe that the number of publications has increased steadily since 2014; although initially most of the papers were published in conferences, over the years

---

[2]https://www.scopus.com/

[3]https://www.sciencedirect.com/

[4]https://dl.acm.org/

[5]The queries were issued last time on April 2021 so some of the results may have changed.

there has been a growing interest in publishing in journals. This figure shows that the problem of POI recommendation is still relevant today. All the works included in our analysis are available as supplementary information[6].

1.3 Contributions of this survey

The purpose of this systematic literature review is to identify the current state-of-the-art in POI recommendation based on LBSNs and to analyze the techniques used, the experimental protocols used to validate them, and the related research challenges. For these reasons, we define the following research questions:

RQ1 What is the state-of-the-art in POI recommendation based on LBSNs? To answer this question we survey the literature in terms of algorithms, information sources, and evaluation methodologies.

RQ2 Which are the most relevant works? We want to analyze with more detail those works that have had more impact in the community, and extract possible reasons for that, exposing these characteristics so that future researchers focus on them in their research.

RQ3 How are these recommenders evaluated? Which protocols/metrics/datasets are used? As a specific goal of this review, we want to dig in the specific evaluation methodologies followed in the POI recommendation literature, since this is a potential source of misbehavior that could limit overall improvements in the field.

RQ4 What are the most important issues to be addressed in the future? Based on the answers to the other research questions, we summarize and present the most important topics that should be considered by the researchers dealing with the POI recommendation problem when using data from LBSNs.

Therefore, our main goal with this survey is to provide a complete review of the works from the last 10 years in the field of POI recommendation based on data coming from LBSNs. As already mentioned before, this is not the first survey that has been done on this subject, however we believe we are the first ones – to the best of our knowledge – that have also considered and, hence, classified the articles by the evaluation protocols followed. The key contributions, thus, of this work are:

- A thorough review of state-of-the-art POI recommendation models based on LBSNs between 2011 and 2020.
- A proposal to classify the algorithmic methodologies used in those works, together with the contextual information handled by the models, and the evaluation methodologies employed to evaluate their performance.
- A list of challenges and open issues in the field, in combination with potential future directions, to help other researchers and practitioners focus their work and resources on the problems that this area needs to fix as soon as possible.

In the next section we present a background on classical recommendation methods and their evaluation, in both cases independent of the domain. Then, in Section 3 we contextualize these concepts to the problem of POI recommendation. Sections 4, 5, and 6 present the main outcomes from our systematic review, first regarding the state-of-the-art algorithms, later about the evaluation methodologies, and finally focusing on the datasets reported in the experiments. We conclude the paper in Section 7 with the most important future research directions and open issues identified after our systematic review.

---

[6]Available here: https://abellogin.github.io/poi_survey/.

2 BACKGROUND ON CLASSICAL RECOMMENDATION

2.1 Problem definition

The main purpose of Recommender Systems (RSs) is to suggest hypothetically relevant items to users. When needed, we denote with $\mathcal{U}$ the set of users in the system and $\mathcal{I}$ the set of items, with $u, v \in \mathcal{U}$ and $i, j \in \mathcal{I}$. Furthermore, since the most typical type of interaction between users and items are ratings, we use $\mathcal{R}$ for the interactions, as it is standard in the area, although other types of interactions exist, such as clicks, buys, watchings, or listenings, depending on the domain [102].

Normally, these algorithms exploit the interactions of the users available in the system to build a model from the data and generate recommendations. Traditionally, the recommendation problem has been defined as an optimization problem [2]:

$$i^*(u) = \arg \max_{i \in \mathcal{I}} g(u, i) \qquad (1)$$

where i^* is the optimal item that maximizes the relevance or utility for user u on any item i among those in $\mathcal{I}$, where such utility function is represented by g. Depending on the domain, items may have different nature, either movies, books, electronic products, or touristic venues, as the focus of this work. At the same time, while the final objective for any of these systems is the same in any case, we classify the most common algorithms used depending on how they work with the data, collaborative filtering and content-based being the two most popular and well-known categories, but other types such as demographic or knowledge-based exist and are applied in the community [102]. In the next subsections, we introduce these two classes of algorithms, together with the most common ways to combine these methods as hybrid approaches; after that, we present basic information regarding how to evaluate recommender systems.

2.2 Content-based filtering (CB)

Content-based recommender systems analyze the items and/or user features (content) and use them to create user and item profiles to recommend items to the target user that are similar to the ones she liked previously. In order to make recommendations, this type of system uses three main components [33]: the *content analyzer* that pre-processes the information available of the items in order to extract keywords, concepts, or other information, the *profile learner* that, using the content information of the items, builds a profile for every user in the system, and, finally, the *filtering component* that matches the user profile against the items in the system.

For modeling the items features from text many content-based algorithms use simple Information Retrieval (IR) models such as the Vector Space Model (VSM) [5], where an n-size vocabulary in the form of keywords or terms is obtained from documents, and then this vocabulary is used to represent those documents in an n-dimensional space. To build the vectors, a common approach is using schemes based on Term Frequency (TF), Inverse Document Frequency (IDF), and combinations thereof (such as the well-known approaches of TF-IDF or BM25) [20]. Once we have transformed all the items into vectors, a similarity metric (such as cosine similarity) can be applied to obtain a ranking of similar items with respect to others the user has previously consumed. Even though modeling this problem with a VSM is still popular nowadays, the use of embeddings has increased lately to exploit possible latent relationships between documents and associated terms [83].

For modeling the users profiles several techniques have also been proposed, including *probabilistic models* (e.g., Naïve Bayes) that will estimate for a target user the probability to classify a document d into class c, that is $P(c|d)$ (e.g., the user likes it or dislikes it, or even one class for each possible rating value), *relevance feedback* that refines the user profile by taking into account their opinion of the previous suggested items and *neighborhood-based* algorithms, where it is common to use a

similarity function computed on the VSM representation of the items and then select the class for the unclassified item taking into account the classes of the nearest neighbor items [33].

2.3 Collaborative filtering

Collaborative Filtering (CF) techniques analyze the interactions between users and items to establish patterns between them when making recommendations. These techniques are normally divided into two groups: memory-based that perform the recommendations using the interactions (usually represented as a user-item matrix) in a direct way by computing similarities between users and/or items [91], and model-based algorithms that build a predictive model by approximating the information stored in the preference or interaction matrix [62]. We now explain some of the fundamental concepts related to these two families of CF algorithms.

2.3.1 Memory-based methods. Memory-based methods (also called nearest neighbors or k-NN) are one of the most well-known and implemented strategies in traditional recommendation due to its ease of programming and the great interpretability of the recommendations obtained [91].

The idea behind these algorithms is to recommend to the target user the most appropriate items by exploiting similarities between the rest of the users/items in the system. For this, they build neighborhoods – by considering those users/items with the highest similarities – and predict the score for new items based on those similarities and the scores provided by such neighbors [91].

Obviously, the similarity function is the most critical component in this type of algorithms, since it is used to select the neighbors and to weight each of them for the final score. Classical similarity metrics exploit trends in ratings such as Pearson correlation or cosine similarity, but recent approaches less focused on the rating prediction problem directly exploit how many items in common are recorded between user/item interactions, by means of variations of overlap measurements such as the Jaccard index [91].

2.3.2 Model-based methods. Model-based algorithms represent the other major family of CF methods, enjoying great popularity because they generally perform better than neighborhood-based models and because of their importance on the Netflix Prize [12]. These models approximate the user-item matrix by transforming both users and items into a latent factor space of low dimensionality so that the user-item interactions can be explained (or recovered) by applying dot products in that space [62].

The most popular method in the area is the standard Matrix Factorization (MF), where the latent space is learned either by applying Stochastic Gradient Descent (SGD) or Alternating Least Squares (ALS) optimization techniques, depending on the domain characteristics and efficiency constraints [62]. However, many other approaches such as PMF (Probabilistic MF), LDA (Latent Dirichlet Allocation), and even the embeddings learned in Neural Networks fit under this family.

Beyond the matrix completion paradigm, several approaches have been proposed to extend this basic formulation to include additional biases and contextual information – like time, sequentiality, and seasonality –, or tags, including well-known techniques like Markov Chains and deep learning techniques [105, 140].

2.4 Hybrid recommenders

Individually, each recommendation algorithm may have some disadvantages in certain situations. For this reason, it is common to combine several models in order to alleviate such problems. For example, CF approaches cannot recommend items to users with very few ratings, while social models need a mechanism to recommend items to those users who have not indicated a social link in the system. There are many ways to make these combinations (we refer the reader to the work [17] for a complete survey about hybrid approaches), although it is usually understood that a

hybrid is any algorithm that combines different sources of information, either explicitly (social and collaborative) or implicitly (two data models generated by different recommendation methods).

Initially, since the most widespread algorithms were CF and CB approaches, hybrid methods combining these two systems proliferated, as in the case of the collaborative via content technique proposed in [6]. However, due to the great expansion of recommender systems other techniques have emerged combining several instances of the same type of recommendation model, like Fossil [53] that combines Markov Chains with similarities models or FPMC [101] that combines Markov Chains with Matrix Factorization.

2.5 Evaluation of recommender systems

The aim of RS evaluation is to determine which recommenders (or configurations of recommenders) are better than others based on the results obtained in certain metrics under a specific evaluation methodology. In fact, among the different types of experiments that can be performed with users of a particular RS – that is, offline, online, and user studies [46] –, the RS community has been mostly focused on offline evaluation, since it is the most comparable across different settings and the one typically used in the literature. In this survey, as we focus on Location-Based Social Networks, offline evaluation will also prevail over online evaluation methodologies since, in most cases, data from LBSNs are readily available and, hence, online studies are not necessary to collect user behavior. However, throughout the rest of this review we will not limit our analysis to this setting, even though it is the most popular one (as we will show later).

For this type of evaluation, the first step is to divide the available data into different sets, so that part of the data is used to build (or train) the recommendation model, while the rest is used to evaluate it (either to validate and test the model in different stages, or using a single withheld subset of the data). The simplest way to do this division is through random partitioning, where a percentage of the interactions is considered for training and the rest for testing (a typical value is 80% for training, thus, leaving 20% for testing). A more elaborate – but quite common – way of doing this partitioning is by n-fold Cross-Validation (CV), where the data is divided into n-disjoint sets, in such a way that $n − 1$ sets are used to build the training set and the remaining one for testing, and this process is repeated n times, so that each set is used once as a test set.

However, these popular random partitioning protocols ignore the temporal component of the interactions, which might be problematic due to the unrealistic evaluation setting [19]. As we shall see in Section 3.4, time-aware splitting protocols are more prevalent in POI recommendation than in classical recommendation, so we defer the explanation of these protocols to that section.

Regardless of how the data is partitioned, we need a way to analyze the performance of the recommenders. Originally, the recommendation quality was equated to how close the recommender was able to predict the rating provided by the user. Hence, error metrics like Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) were used, however, since these metrics only account for the observed items, they do not reflect well real-world problems nor the perceived user experience [87]. Because of this, IR ranking metrics like Precision, Recall, or nDCG (normalized discounted cumulative gain) were used to measure how many relevant items were included in the ranking generated by the RS [13].

Moreover, despite the importance of relevance in recommendations, there has been a growing awareness on measuring other evaluation dimensions like novelty (as opposed to recommending popular items) and diversity (accounting to how many items with different features are recommended), as sometimes producing only accurate recommendations may not surprise or discover new items to the user [22]. Nonetheless, it should be noted that, even though it is advisable to have good recommendations in all these evaluation dimensions (i.e., novelty, diversity, accuracy, etc.),
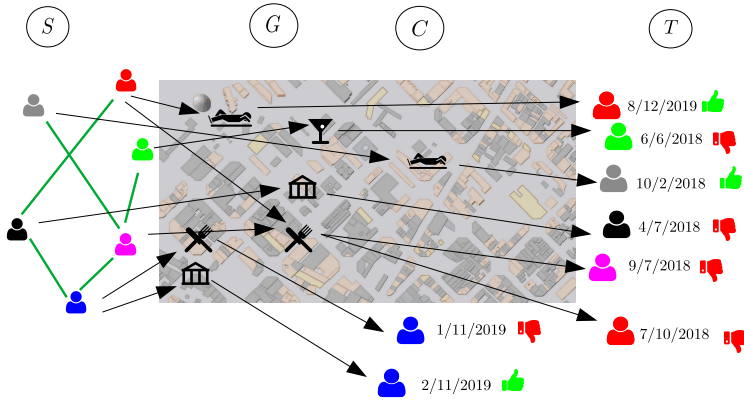
Fig. 2. Graphical representation of the data found in a LBSN. Each letter represents one typical information source available in such data, with S showing the social relationships between the users, G and C show the geographical (physical coordinates in the map) and categorical (the type of POI) influences of the POIs, and T shows the moment in time when the user visited the POI and her opinion in the POI (temporal and textual/rating information).

it is in general difficult to find an algorithm that outperforms any other method in all possible situations [103].

3 POINT-OF-INTEREST RECOMMENDATION

3.1 Problem definition

The key concept in POI recommendation is to suggest users new places to visit when they arrive to a city or region, like museums, restaurants, or hotels. Location-Based Social Networks (LBSNs) shape the data used by most of the literature devoted to this problem, and in particular, by those works analyzed here. In these social networks users may establish social links with other users in the system, share information, and record check-ins to the specific venues they visit when located in a city.

Figure 2 depicts the different types of information that can be stored in and collected from these LBSNs. Due to the great wealth of information available on these social networks, several recommendation objectives have been defined, including recommending locations, trips, activities, or friends. As we have already introduced previously, in this review our focus is on the problem of POI or venue recommendation, for a review oriented on the rest of recommendation objectives, we refer the reader to the survey of [36] and those discussed in Section 1.1.

Let us formalize the problem of POI recommendation. In order to help the reader throughout the rest of the document, we will adapt the notation used in Section 2.1 as follows. Since in this case the items are POIs (locations) and the ratings are check-ins, we will use the letter $\mathcal{L}$ to denote the POIs and letter C to denote the check-ins as in [81]. Moreover, even though the POI recommendation task is similar to the classical recommendation problem, it has some particularities that differ from the traditional recommendation. These include but are not limited to:

- Sparsity: normally, the sparsity (the ratio between observed and potential preferences) is very high. For example, the density of the Netflix and Movielens20M datasets (used in classical recommendation) are approximately 1.77% and 0.537% respectively, while the density of datasets from Foursquare [122] and Gowalla [29] are 0.0034% and 0.0047% respectively.

- External influences: while in classic recommendation the only information usually exploited is the user-item matrix (user, item, score, and sometimes the timestamp associated), POI recommendation is highly affected by geographical (coordinates of the visited venues), social (friendship relationships between users), and temporal (specific moment in time when the user visited the venue) influences. Even if the use of all these influences is useful in this type of context due to the high sparsity, the geographical influence is possibly the most important aspect to consider. As the Tobler's first law of geography states [88]: "Everything is related to everything else, but near things are more related than distant things". These influences are not only important to improve the performance of the algorithms, sometimes it is mandatory to take them into account because they impose certain restrictions on the recommendations. For example, some POIs such as shops, restaurants, museums, etc. are only open for a certain period and users cannot make visits to POIs that are too distant from each other.
- Implicit information: in classic recommendation the information encoded in the user-item matrix has been traditionally modeled using ratings. However, in most POI recommendation datasets (e.g., Brightkite, Gowalla, or Foursquare), we only have the specific moment in time when a user visited a POI. In fact, the users may have checked-in more than once in the same POI (something that it is not possible in classic recommendation). In order to model these repeated preferences, researchers build frequency matrices in which each entry represents the number of times a user checked-in in a venue.

Considering these features, for POI recommendation, Equation 1 should be replaced by a more appropriate one as follows:

$$l^*(u) = \arg \max_{l \in \mathcal{L}} g(u, l, \theta) \tag{2}$$

where in this case θ represents a contextual variable (e.g., the geographical information of the POIs and users, temporal influence, social context, etc.).

In the next subsection (Section 3.2), we present in more detail the different information sources that are usually exploited in POI recommendation. Then, in Sections 3.3 and 3.4 we characterize the particularities of the algorithms and evaluation methodologies, respectively, when applied to this problem. Finally, Section 3.5 presents the relation between this and other recommendation tasks.

3.2 Alternative information sources

As presented before, the density of most POI recommendation datasets is very low. For this reason, the vast majority of the analyzed POI recommendation approaches use more than one source of information. These include:

3.2.1 Interaction types. Although we have equated check-ins in LBSNs with the main interaction between users and items (as ratings in classical recommender systems), this is not the only type of interaction recorded in this type of systems. Other LBSNs – such as Yelp – allow users to perform reviews of the POIs they visit and, in some cases, rate the venue; through these reviews we determine whether the user liked the POI or not, either by directly considering the rating or by analyzing the sentiment of the text in the review. Other works obtain the items to be processed from the photos that users take and upload to other applications such as Flickr or Instagram [90, 116], which may include GPS coordinates as their metadata along with visual information, so that the path followed by the users could be recovered. Similarly, user generated content tagged with GPS coordinates – such as tweets from Twitter or the traces left by mobile apps – can potentially be used in POI recommendation applications.

3.2.2 Rich side information of items. The items in this type of systems, Points-Of-Interest, can be associated with a richer kind of information than in other domains. First, each POI has a geographic

location associated, although this information is not always available in the datasets. This source of knowledge is especially relevant because people tend to go to places that are close to each other. Sometimes this information is exploited to calculate centroids or clusters of activity for either users and items in order to make recommendations [82, 107]. We consider that an algorithm uses this kind of information if it uses the user/POIs coordinates somewhere in the proposed model (e.g., when computing distances, creating clusters, building distributions based on proximity, and so on).

Second, and more similar to the traditional recommendation situation where items usually have associated characteristics – such as genres in the movie, book, or music domains –, in POI recommendation the venues are frequently linked to a specific POI category (for instance: restaurants, hotels, parks, museums, etc.), which may have different levels (thus, building a category hierarchy) depending on how specific the category is – for instance, a venue could be labeled as a *Vietnamese restaurant*, an *Asian restaurant*, or simply as *Food*. This information is very useful and, as we will show later, exploited in many works [131, 136], since some users may be more interested in visiting only certain types of POIs while, at the same time, it is not very common for a user to visit very similar POIs all the time, affecting the recommendations.

On top of this, we may find approaches that make use of the opening and closing times or the time windows or prices of the POIs, since these are particularly important characteristics when creating practical recommendations for users of real systems. However, it should be noted that this type of information is generally considered in works that are evaluated with user studies or mobile apps, or that try to solve a different problem where constraints on the recommendations need to be taken into account (for example, trajectory instead of POI recommendation), and hence, they are less represented in this review because some of those approaches are out of its scope.

3.2.3 Textual reviews. In some LBSNs, users can not only register their check-ins, but also write reviews about the POIs they have visited and exchange this information with other users of the system, either as long, more elaborated texts (as in Yelp) or as short, concise texts (as the so-called *tips* in Foursquare). This type of textual information can be exploited by recommendation approaches and structure this information using topic modeling techniques like Latent Dirichlet Allocation (LDA) or Latent Semantic Analysis (LSA) [99]. This textual information may provide more useful and high-quality information about the users' interests since, in combination with check-in data, it is possible to capture when the user visited a venue and whether she liked it, together with the reasons about such opinion.

It is important to note that, as the textual information available from reviews is different than the aforementioned POI features (since such features are intrinsic and static to the items, they do not change, while the reviews represent a subjective opinion from the user perspective), in our classification we will make a distinction between these two types of information, counting differently those works that exploit textual reviews or POIs features. However, it must be taken into account that both textual and content information are related, since in some cases the researchers work with content information obtained from the text, as in [49, 86, 139].

3.2.4 Social links. As we already know from other domains, users tend to be more interested in a product when their friends have some opinion about it; in the same way, this type of information may influence users when receiving POI recommendations. Because of this, some approaches exploit such social links when predicting the user preferences, for instance, by replacing the collaborative neighborhood in classical CF methods with those users who have some social relation with the user [28, 127], or by building social graphs between the users in the system [115].

It should be considered, however, that the social links that exist in LBSNs, even though they are usually denoted as "friends", because of the nature of these networks, it is very likely that they do not correspond to friends outside of the system, but similar-minded people or with close tastes,

interested in following their opinions or controling the places they visit. In fact, some datasets that include this information it is actually extracted from a different social network (for instance, the global-scale dataset from Foursquare reports friends from Twitter [122]), so these social links should be exploited with great care.

3.2.5 Sequential and temporal information. As discussed before, the temporal dimension is essential in the domain of POI recommendation, mainly because it affects significantly the type of venues that can be visited, but also because users tend to diversify when deciding the next place to visit. Hence, it becomes paramount to know, and to consider in the recommendation process, the users' previous visits. Similarly, since the user interactions usually have a timestamp associated, it is possible to exploit this data to know the evolution of users' tastes over time; it can also be used to detect the periods of time where some POIs have more activity than others (e.g., bars and restaurants from midday onwards).

In this survey we consider that an algorithm uses sequential information if it processes or analyzes the different events when they occur immediately one after the other or if they exploit successive visits to different POIs. At the same time, we assume that a model uses temporal information if they work with the different timestamps of the check-ins or if they use time schedules of the POIs.

It is important to note the distinction between temporal and sequential information. While these are clearly related terms, they are not completely equivalent: not every sequential event need to be temporal and vice versa. For example, once we know a user visited three venues at 4PM, 8PM, and 10AM, we might be tempted to create a sequence of length 3, however, it is very likely that the user stopped to rest during the night, so the sequence should be splitted; the inverse case is more obvious: if we know the sequence followed by a user, it is impossible to recover the exact timestamps unless we know information about the initial time, and the time involved to go from each venue to the next, together with how much time was spent in each of them.

3.3 Characterization of POI recommender systems

In this section, we classify existing research works according to six main classes of algorithms, based on the most frequent approaches we have identified in our analysis: based on similarities, factorization models, probabilistic approaches, deep learning techniques, graph- or link-based methods, and hybrid models. These categories may or may not use more than one information source among those presented in the previous section, as we shall discuss in detail in Section 4. In the following, we describe these categories together with some representative methods from the state-of-the-art reviewed in this survey.

3.3.1 Based on similarities. These algorithms correspond to the classic k-NN approaches explained in Section 2.3.1, where researchers use similarities between users or items like the well-known cosine similarity. The pure user-based CF approach is defined as follows:

$$\hat{g}(u,l) \propto \sum_{v \in \mathcal{N}_l(u)} sim(u,v)c_{vl} \tag{3}$$

where $\hat{g}(u,l)$ represents the predicted score for the user-location pair (as in Equation 2), c_{vl} indicates the influence of venue l on user v (usually as a function of the check-in frequency), $\mathcal{N}_l(u)$ denotes the neighbors of user u that have also visited location l, and $sim(u,v)$ represents the similarity between users u and v.

Due to the additional information available in this domain, some authors incorporate a temporal decay in the formulation or even use similarities based on the geographic distance between items. For example, the UTE+SE approach from [134] divides the check-in matrix in different time slots

and uses it in a user neighborhood CF model; however, since this increases the data sparsity, the authors add a term in the prediction score to account for the similarity between time slots.

However, as social links between users are often available, instead of calculating similarities between users, in some works, they use the friends of the target user as "nearest neighbors" as they assume that friends in this type of networks may have common interests, as done in the MLR model from [45].

It should be noted that in the examined works, neighbor-based models are usually an intermediate phase of a more complex algorithm. That is why we decided to extend the category beyond k-NN approaches to consider any proposal that use similarities between items/users and/or use these similarities to establish relationships between them. As a particular example, the LARS approach proposed in [64] would fit in this category, since it takes into account two different similarity spaces: preference locality (users in the same region tend to have similar tastes) and travel locality (users tend to travel short distances when visiting the venues of a region).

3.3.2 Factorization. The basic premise of this family of algorithms is to decompose the check-in matrix $C \in \mathbb{R}^{|\mathcal{U}| \times |\mathcal{L}|}$ into two matrices, one for users $\mathcal{U} \in \mathbb{R}^{|\mathcal{U}| \times |K|}$ and one for POIs $\mathcal{L} \in \mathbb{R}^{|\mathcal{L}| \times |K|}$, with K being the number of latent factors, Formally, these models try to optimize the following function:

$$\min_{U,L} ||C - UL^T||_F^2 + \lambda_1 ||U||_F^2 + \lambda_2 ||L||_F^2 \tag{4}$$

However, in most models, the previous formulation is augmented by incorporating additional influences such as geographical or temporal ones. We want to note we did not name this class of techniques as the most frequent name *matrix factorization*, because algorithms using tensor factorization (where the additional dimension is used to model time or geographical information) or other types of latent factor models also fit in this category.

Recommendation approaches that belong to this type include GT-BNMF, proposed in [74], which is a geographical probabilistic factor analysis framework that takes into account the geographical influence and the textual information of the POIs, to avoid limitations from pure collaborative information such as the cold-start problem. In fact, factorization approaches that exploit the geographical information are very frequent in this domain, as this is a critical information source. The following three models have become state-of-the-art baselines because of their popularity in the area. First, GeoMF, a weighted factorization model proposed in [73] that divides the full geographical space into different grids to model the following influences: user activity areas and POI influence areas. Second, IRenMF from [82] incorporates geographical information in the form of neighboring POIs of the target item by exploiting two types of influences: the instance level influence (assuming users tend to visit neighboring locations) and region level influence (to capture user preferences that are shared in the same geographical region). Third, in RankGeoFM from [68] the authors propose a geographical factorization method that incorporates the influence of the neighboring POIs of the target item by including a distance weight in the optimization formula.

Other methods like GeoIE proposed in [114] also incorporate geographical influence, but in this case a power-law distribution is used to consider that POIs that are far from other POIs in the system are less likely to be selected. A tensor model is introduced in [52], where the authors apply factorization techniques to transition tensors so that transitions between consecutive POIs are modeled, together with a geographical preference term so that far away POIs are less likely to be selected, just as in the previous approach. Finally, SPR from [142] also fit in this category. In this model, the authors incorporate the geographical influence (distance between POIs and users) and sentiment similarity between POIs extracted from micro blogs into a classical MF approach.

The temporal dimension is exploited in [40], where the authors propose LRT, a matrix factorization model that incorporates the temporal effects of the POIs by considering two properties: non-uniformness (the users have different preferences during the day) and consecutiveness (users tend to have similar preferences in consecutive hours). STELLAR, the model proposed in [146], is a time-aware successive POI recommendation model by using a four-tuple tensor while adapting the Bayesian Personalized Ranking (BPR) optimization criteria from [100]. Geo-Teaser as proposed in [145], on the other hand, combines two different models: a temporal POI embedding for sequential influence that differentiates between weekday and weekends, and a hierarchical pairwise preference ranking model based on BPR that discriminates POIs based on the distance between them.

Social information has also been used in factorization techniques. For instance, TenMF from [126] is a tensor factorization approach (integrating users, venues, and time frames) that incorporates spatial and social influences in the regularization terms. GeoEISo is an MF approach based on the SVD++ model proposed in [43] that incorporates both geographical and social influence (in particular, the trust relationships between the users). The model TGSC-PMF proposed in [99] also combines different information sources, since its probabilistic matrix factorization component exploits categorical and textual information by using an LDA technique, a kernel density estimation uses geographical information, and social information is combined through a power-law distribution.

Categorical or content information, as in the last method described, is easy to be integrated in factorization methods. For instance, CAPRF is proposed in [41] where besides the user and POI latent matrices, it incorporates the content and sentiment analysis obtained from the user tips. In a more complex method, ASMF merges social, geographical, and categorical influences, by exploiting check-ins of social, location, and neighboring friends in order to learn the potential locations to recommend, and using an additional score based on a distance distribution between the users's home and their actual check-ins to model the geographical influence, while the categorical information is considered through an additional weight in the recommendation score.

3.3.3 Probabilistic. Probabilistic approaches typically consider several random variables that might be related according to some laws or formulations, which in recommendation usually involve users, items, and the potential interaction between the former and the latter. Probabilistic graphical models are one of the most useful frameworks that allow to encode these probability distributions over arbitrary domains, however it is possible to also define simple probability models just by applying Naïve Bayes or other simple approximations with strong (and probably not too realistic) independence assumptions. Besides those techniques that match these formulations, we also extend our categorization as probabilistic to any model that uses some kind of probabilistic distribution in its algorithms to represent or process the data.

In this sense, for example, we consider that those approaches that model the geographic influence by means of power-law distributions such as [114] and [99], those that make use of the Kernel Density Estimation (KDE) like [137], or those using Bayesian algorithms in the inference or in the optimization steps as in [69] fit into this category. Another example can be found in WWO from [80], which is a model that exploits the sequential preferences of the users to recommend POIs within a time duration; for this, it estimates the distribution of the temporal intervals and creates a low-rank graph to deal with the sparse conditions of the data.

It is important to mention that many proposals can be classified as members of the probabilistic and factorization categories, such as Probabilistic Matrix Factorization (PMF) or some formal topic modeling algorithms, like Latent Dirichlet Allocation (LDA); some examples can be found in [48, 99]. For instance, Poisson Geo-PFM, the algorithm proposed in [77], is a geographical

probabilistic factor method that models the geographical influence by using a parametric power-law distribution to represent the users activity areas over a set of latent regions, and HI-LDA [120] and MMBE [57] are latent probabilistic models based on LDA. HI-LDA exploits three different factors: community-behavior (social information), region-POI component (geographical information) and the sentiment-word (textual data), while MMBE is a multi-modal Bayesian embedding model that exploits several influences: social (using user embeddings), sequential (using skip-gram and DeepWalk, a mechanism to learn embeddings of vertices in a graph [95]), geographical (exploiting different regions), content (topics) and temporal (used for modeling the distribution over topics). A different approach is GAIMC, a method that first models the geographical influence by using a Gaussian Mixture Model (GMM) and then uses a matrix completion approach to perform the recommendations.

In the same way, we consider proposals based on Markov Chains (MC) as probabilistic since they model the probability of going to the next POI using the immediately previous visited POIs. In fact, this is one of the most popular approaches because of its simplicity and expressiveness. For example, the authors of [27] propose FPMC-LR, an approach that makes use of Factorized Personalized Markov Chains (FPMC) but adding physical restrictions: instead of building the entire transition tensor, only neighbor POIs are considered after dividing the Earth in different grids; then, a modified version of the BPR optimization technique is used to take into account the sequential components. PRME-G is a next-POI metric embedding method proposed in [38] that models the sequential influence by borrowing ideas from Markov Chains: instead of computing the transition probabilities by counting, they are estimated by computing the Euclidean distance of the POIs in a latent space.

A more complex method is proposed in [129], where the approach called MEAP-T considers the sequential component between the POIs (using a first-order MC) and also the temporal influence by modeling the periodicity and the time intervals between the POIs; then, the user preferences, POI transitions, and POI and temporal relationships are transformed into three latent spaces, while exploiting the Euclidean distance and using BPR as optimization criterion.

3.3.4 Deep Learning. Deep Learning (DL) encompasses a set of techniques from the Machine Learning area. While they emerged throughout the 20th century, in the area of recommendation their popularity has begun to grow in the last 10 years. When processing and learning from the data, these types of techniques make use of layers of artificial neurons in order to obtain different representations of the data by optimizing a differentiable function. Although there are many types of neural networks, some of the best known are [140]: the Multilayer Perceptron (MLP), that is the most basic neural network composed by one or more hidden layers between the output and the input layer using different activation functions in each neuron; the Autoencoder (AE) and Variational Autoencoder (VAE), that are unsupervised techniques oriented as compressing and then rebuilding the original data (VAE also assumes that the input data follows a probability distribution and tries to learn the parameters of that distribution); Convolutional Neural Networks (CNN), oriented at processing images using pooling operations and convolutional layers; and Recurrent Neural Networks (RNN) that memorize previous computations for processing sequential information. As we shall see later, these approaches for POI recommendation have become paramount in the last 4 years (and in the year 2020 it has been the most extended type of model in the area); some paradigmatic examples are the following.

First, PACE is a deep learning technique proposed in [121], where an architecture with three main components is presented: an embedding layer that takes as inputs the embeddings of the POI and user, the context layer used for context prediction, and the preference layer composed by multiple feed-forward layers. Second, VPOI from [116] is one of the few approaches that use

images for POI recommendation, because of this, here the authors use CNNs to extract the visual contents from the images and which are later exploited in the learning process. Other examples are CARA, an approach based on RNNs proposed in [85], that consists in two gating mechanisms: the first one to control the influence of ordinary contexts and the second one to model the sequential influence by analyzing time intervals and geographical distance between successive check-ins and SAE-NAD, an autoencoder model in which the encoder uses a self-attentive mechanism in which the most representative POIs contribute more to the hidden representation of the user, while the decoder incorporates geographical influence with a radial basis function kernel using the pairwise distance between the POIs. Finally, STGN from [143] is a spatio-temporal gated network model that incorporates two temporal and two distance gates to LSTM, to control the influence of short (recent visited POIs) and long (all previous visited POIs) term preferences.

Other deep learning techniques that have been used more recently in the area of POI recommendation are embeddings, specifically graph and word embeddings. The latter consists of learning a latent representation of the words so that those that have a similar meaning also have a similar representation [89], while in graph embeddings the objective is to transform a graph into one or more d-dimensional vectors which preserve the graph information as much as possible [18]. Nevertheless, other techniques such as matrix factorization are used to learn these embeddings, so in this review we will include these proposals within the family of deep learning techniques or factorization depending on the specific case. One example from the POI recommendation domain is STA from [96], where the authors define a graph embedding approach that incorporates both temporal and geographical information.

3.3.5 Graph/Link. Link-based or graph-based techniques build one or more graphs using the data stored in the system, which in our case is a LBSN. They typically consider the users or POIs as nodes, and exploit various influences (e.g., geographical, social, temporal, etc.) to create and weight links between these nodes. There are a great number of models based on graphs, among which the following are the most commonly used in POI recommendation approaches: Random Walk, Hypertext Induced Topic Selection (HITS), PageRank, etc. However, as we shall see in the next sections, its popularity in the area of POI recommendation is not very high.

Among the few (representative) examples we have found in the literature, the following is a paradigmatic example of how this type of methods are used. In [92], a Random Walk approach is proposed, where a graph is built in which both the venues and users are nodes of the graphs, and where a link exists between a user and an item whenever the user has checked-in in that item; additionally, users are linked to each other based on their social relationships. In the model proposed in [135], GTAG, also two types of links are used, but considering different information: geographical and temporal influence; on the one hand, POIs are connected by distance to the nearest venues and weighted according to a power-law distribution, on the other hand, users and POIs are connected according to sessions defined based on their check-ins and using an exponential function to weight the edges to account for the temporal influence; with all this information, a Breadth-first Preference Propagation algorithm is used.

Thanks to the flexibility of these models, they can exploit almost any type of information source. For instance, in [7] the authors propose an online POI recommendation model where a weighted categorical tree is built for each user, where a HITS-based approach is used to obtain local experts, which are later used to produce recommendations. A more complex approach is found in UPOI-Walk, where a Dynamic HITS-based Random Walk model is proposed in [130] that combines several relationships captured in the data: popularity (between POIs and check-ins), social (between POIs and users' social circles), and categorical (between semantic labels and user preferences).

3.3.6 Hybrid. Contrary to the more classical understanding of how hybrid methods are defined [17], in this review we do not classify approaches using and combining several components within the same algorithm as such – for example, adding a similarity computation in a matrix factorization formulation or using a matrix factorization algorithm in a more complex deep learning model. We decided to follow this procedure because, as we have discussed previously and it can be observed in most of the examples shown before, most algorithms combine several sources (geographical, temporal, sequential, social, etc.) in different ways, in such a way that if we took the more traditional and strict definition of hybrid recommender, almost every recommendation approach could be considered as such. Hence, most of the analyzed hybrid approaches follow this formulation:

$$\hat{g}(u, l, \theta, H) = \sum_{h=1}^{|H|} w_h \cdot \hat{g}_h(u, l, \theta) \qquad (5)$$

where θ represents again the contextual information, whereas H denotes the set of components of the hybrid approach and w_i describes the weight associated to the corresponding component. Note that sometimes the aggregation function is not a sum but a product operation.

In the following, we present some of the approaches that we do consider as hybrids, starting with those that integrate social information, since it was identified in many hybrid methods. One possible reason for this is that this information source cannot be easily modeled under a unified framework together with other sources due to its different nature, hence, it needs tailored combinations or aggregations as the ones we present next. For example, the UPOI-Mine approach proposed in [131] besides considering the individual preferences of the users (through the tags of the previously visited POIs), it exploits the social information from the target user friends and uses the popularity of the venues in order to counter the data sparsity; all of this is then combined into a regression tree model, focused on predicting the next restaurant to visit, instead of general POIs. On the other hand, the USG model proposed in [127] combines three different scores: user preferences, social information through a combination of the classical user-based CF formulation, and geographical influence with a power-law distribution. Similarly, various information sources are combined in [45], although in this case the authors model POI recommendation as a multi-objective optimization problem considering social, geographical, and user similarity influences.

As in the previously described method, we found several approaches where two other sources of information besides social are exploited. LORE is a method proposed in [137] that combines social (by computing similarities between friends), geographical (using a two-dimensional Kernel Density Estimation), and sequential (with an additive Markov Chain trained with the transition probabilities between all the users) information. Categorical information is exploited in GeoSoCa, a model proposed in [136] where social, geographical, and categorical influences are combined, using a power-law distribution for the first and last models, whereas a similar method to the one described in LORE is used for the geographical one.

Besides social information, geographical (as already discussed in other parts of this review) is another source that is exploited frequently; indeed, this is evidenced in the following hybrid approaches which exploit this type of signal among others. In [79], the authors propose a hybrid model that combines a matrix factorization approach using a transition matrix to model the transitions between POIs with a power-law distribution. The so-called LBPR method from [51] adapts the BPR technique to predict the next category and then obtain a ranking of POIs using the predicted category and after incorporating a geographical score for the candidate POIs; the main difference with other approaches based on BPR is that this model uses lists of categories instead of category pairs in the learning step. The APRA-SA model as proposed in [107], on the other hand, takes into account geographical and temporal information by computing the popularity of the

Table 3. Evaluation protocols characterized by their application level and the type of split; where X+Y denotes the type (X, either <u>R</u>andom or <u>T</u>emporal) and application level (Y, either <u>S</u>ystem or <u>U</u>ser) of the split, and ✓, ✗, and **?** indicate if that characteristic is known to occur, never occur, or it is impossible to know in a protocol.

Characteristics	R+S	R+U	T+S	T+U
All users are evaluated	**?**	✓	**?**	✓
Leaks future data in test from training	Very likely	Very likely	✗	For some users
Test contains recent check-ins	**?**	**?**	✓	✓
Test is made as a random subset	✓	✓	✗	✗

POIs in different time periods and using a Kernel Density Estimation component. Finally, the GE method proposed in [119] consists of a graph-based embedding model where four types of graphs are considered: a POI-POI graph (to capture the check-in sequences of POIs), a POI-region graph (to exploit the geographical information), a POI-time graph (for temporal and cyclic behaviour), and a POI-word graph (to exploit semantics). Finally, TECF from [118] is a hybrid approach that combines user-based collaborative filtering (based on DeepWalk), a temporal user-based collaborative filtering, and a power-law distribution for modeling the geographical influence.

3.4 Characterization of evaluation methodologies

The evaluation methodologies used in the POI recommendation domain are not too different from those traditionally used in classical recommendation and presented in Section 2.5. However, considering the importance some dimensions have in this domain – i.e., time and geographical information, mostly – we describe now in more detail those time-aware evaluation methodologies used in the area [19], together with some variations inherent to the POI recommendation problem.

As explained in Section 2.5, the first step in any offline evaluation is to divide the available data into different sets: at least training and test, although an additional validation set is preferred to tune the parameters of the models and not overfit the test set. How the original data is divided is critical to imitate the use of the recommendation algorithm in a real scenario, that is why in the POI recommendation task we observe that the temporal dimension is often used when splitting the dataset, even though these methodologies were already used and formalized in the area, nonetheless, random partitions of the data are still very popular [19].

More specifically, we consider a split is *temporal* whenever the check-ins are ordered according to the temporal dimension (either by the actual timestamps or because there is some sequential information in the data) so that the check-ins in the test set are more recent than those in the training set; otherwise we consider the split is *random*, including the cross-validation setting presented in previous sections. An additional criteria that may have a great impact on the final results is whether the split is done at the system or user level; this reflects whether the previous criteria (temporal or random) is applied to the whole dataset or in a user basis – i.e., for each user independently. This criteria affects the number and type of users that belong to each set, since performing the split at the user level would guarantee that all the users exist in both training and test sets. Finally, a parameter that defines different variations of these protocols is whether the subset is selected according to a percentage or ratio between training and test – typical values are 80% for training and the rest for test – or based on a fixed number of elements, for example, the last 1 or 2 interactions go into the test set; for the sake of a cleaner presentation we will not consider this parameter in the classification we use in the next sections. We present in Table 3 a summary of the implications for the four possibilities regarding data splitting that will be used in the rest of the paper. Even though we have found some papers where the evaluation was performed in other

ways (e.g., temporal windows or splits by distance between locations), most of the analyzed articles fit into the aforementioned evaluation protocol classification. Based on this, we argue that the most realistic scenario is a temporal partition at the system level, as it takes into account the temporal dimension while avoiding any leaking of the user interactions from the future into the training set.

Because of the paramount importance of the geographical dimension, some authors include in their experimental settings variations tailored for the POI recommendation problem. In particular, those approaches that exploit geographical information or neighbor venues tend to filter the data by cities or, in general, by geographical regions, such as country or continent. Another important characteristic of the data produced by LBSNs is that users may visit more than once the POIs, hence, this leads to two ways of producing the splits explained before: at the check-in level (keeping all the check-ins, even the repeated ones) or at POI level (removing the duplicated user-POI pairs and keeping only one instance before running the splitting strategy). These repetitions may have a significant effect in training since it allows to capture item frequencies at the user level, but its effect is even more dramatic in the test set, since it may hide the fact that some uninteresting baselines (such as returning those items previously interacted by the user) would perform very well [104].

Finally, regarding the metrics used when evaluating POI recomender systems, it should be noted that error-based metrics are not very interesting when the interaction to be predicted is a check-in, since that value is always 1; when the user interaction is different (such as ratings or those described in Section 3.2.1), then these metrics can be applied, considering the limitations already described in Section 2.5. Nonetheless, it is important to mention that in recent years, researchers dealing with the problem of POI recommendation and related tasks (see Section 3.5) have adapted ranking metrics to consider distances between the recommended POIs and the actual order followed by the users in the test set; some examples can be found in [23], where the authors use a metric based on F_1 that takes into account the pairwise order between POIs, and [104], where the Longest Common Subsequence algorithm is introduced in ranking metrics to penalize those recommendations less similar with the sequence followed by the user.

3.5 Relation to other recommendation tasks

POI recommendation is not the only task that can be performed using data from LBSNs, due to the richness of the data of this kind of social network, a large number of related tasks/problems have arisen. Since they are not the focus of this review, we discuss them briefly now:

- Trajectory (or route) recommendation: typically, POI recommendation approaches provide each user with a list of POIs that hopefully may be of interest; however, there is generally no intrinsic relationship between these recommended POIs. Instead, in route recommendation, a complete trajectory is generated and provided to the target user. Because of this, additional restrictions must be taken into account, such as the duration or length of the route or the schedule of the venues [23].
- Friend recommendation: this is a well studied problem in the context of traditional social networks, like Twitter or Facebook. Considering the importance of the social dimension in LBSNs in general, and of social information for POI recommendation (discussed in previous sections), there is an increasing interest in this problem by the community.
- Group recommendation: quite frequently users visit a city in groups, either composed by friends, family, or even as organized tours. In this case, instead of recommending POIs to a unique target user, the algorithms should be tailored to groups of users. This problem needs to take into account additional factors, such as the difference between passive and active users in such groups, or the balance between individual preferences.
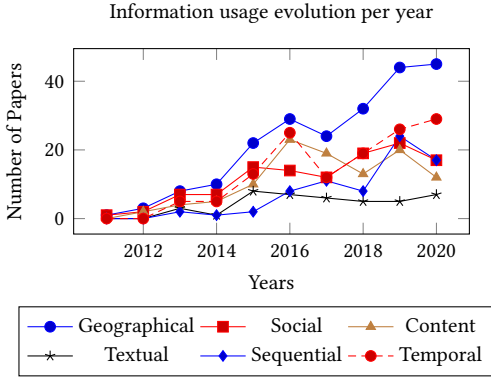
Fig. 3. Number of papers using different information sources by year.
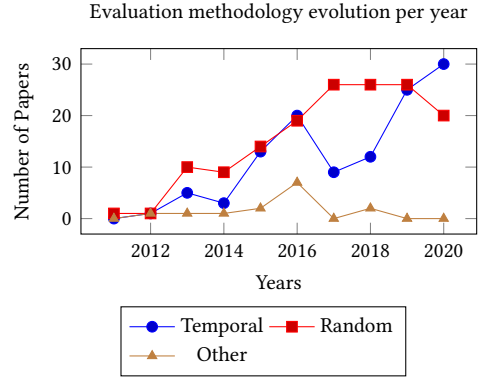
Fig. 4. Number of papers using different evaluation methodologies by year.
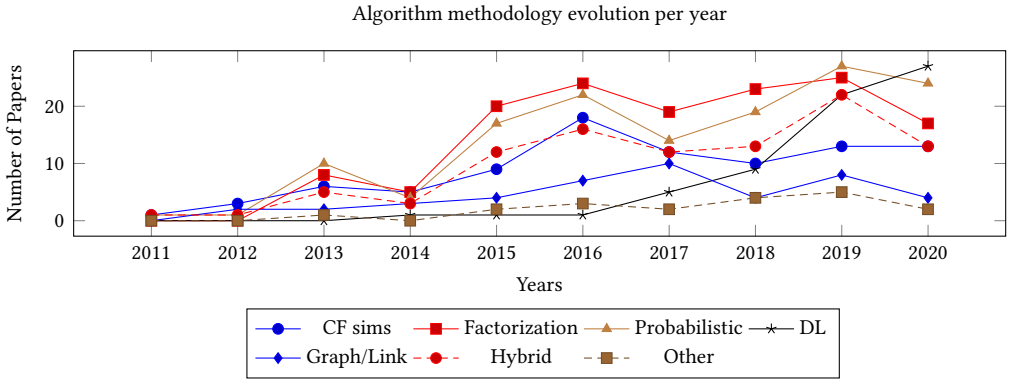
Fig. 5. Number of papers using different algorithms by year.

4 SYSTEMATIC REVIEW OF STATE-OF-THE-ART ALGORITHMS

In this section, we analyze the state-of-the-art algorithms according to the classification presented in Section 3.3 for the papers considered in this study (described in Section 1.2). Since the number of papers included in this review is very large (more than 300), we selected the *most representative* papers for each year and include their whole characterization in Table 4. When selecting the most representative papers, we considered the top-5 most cited articles per year according to Scopus with at least one citation[7]. We also include in the table two summary rows that count how many papers (among the sets of most representative or the entire collection) satisfy each condition. Each of the conditions (columns) correspond to the categories described in Sections 3.2, 3.3, and 3.4, respectively.

Based on this table, let us first analyze the trends on information sources used throughout the years. We observe that most of the algorithms make use of geographic information in some way (either by calculating distances between POIs, grouping users and POIs in clusters according to regions, modeling movement distributions, etc.). Most researchers argue that this type of information is critical since users tend to visit POIs close to where they are and this conclusion can be obtained

---

[7]The number of cites reported have been obtained on April 14, 2021.

Table 4. Summary of analyzed POI recommendation approaches sorted by publication year. The ✓mark denotes that the proposed model has the feature indicated in the column, whereas (N.A.) shows that no acronym was given.

			Information used						Model							Split type		
Year	Reference	Acronym	Geographical	Social	Content	Textual	Sequential	Temporal	CF sims	Factorization	Probabilistic	DL	Graph/Link	Hybrid	Other	Random	Temporal	Other
2011	[127]	USG	✓	✓					✓		✓			✓		✓		
2012	[64]	LARS	✓						✓							✓		
2012	[7]	(N.A.)	✓		✓				✓					✓				✓
2012	[131]	UPOI-Mine	✓	✓	✓				✓					✓				
2012	[92]	RW, Weighted-RW		✓							✓		✓				✓	
2013	[79]	(N.A.)	✓		✓		✓	✓		✓	✓			✓			✓	
2013	[74]	GT-BNMF	✓			✓				✓	✓					✓		
2013	[27]	FPMC-LR	✓				✓			✓	✓						✓	
2013	[40]	LRT						✓		✓						✓		
2013	[134]	UTE+SE	✓					✓	✓		✓			✓		✓		
2014	[130]	UPOI-Walk	✓	✓	✓								✓					
2014	[135]	GTAG	✓					✓					✓			✓		
2014	[73]	GeoMF	✓							✓						✓		
2014	[82]	IRenMF	✓							✓						✓		
2014	[137]	LORE	✓	✓			✓		✓		✓			✓			✓	
2015	[77]	Poisson Geo-PFM	✓							✓	✓					✓		
2015	[68]	RankGeoFM	✓					✓		✓							✓	
2015	[136]	GeoSoCa	✓	✓	✓						✓			✓			✓	
2015	[38]	PRME-G	✓				✓	✓		✓	✓						✓	
2015	[41]	CAPRF				✓				✓						✓		
2016	[119]	GE	✓		✓		✓	✓		✓	✓			✓			✓	
2016	[65]	ASMF	✓	✓	✓					✓			✓				✓	
2016	[146]	STELLAR						✓		✓							✓	
2016	[52]	(N.A.)	✓				✓			✓	✓						✓	
2016	[80]	WWO					✓	✓		✓							✓	
2017	[145]	Geo-Teaser	✓				✓	✓		✓	✓	✓					✓	
2017	[121]	PACE	✓	✓								✓					✓	
2017	[99]	TGSC-PMF	✓	✓	✓	✓				✓	✓					✓		
2017	[51]	LBPR	✓		✓		✓			✓	✓			✓			✓	
2017	[116]	VPOI		✓						✓	✓	✓				✓		
2018	[84]	SAE-NAD	✓									✓				✓		
2018	[85]	CARA	✓				✓	✓				✓					✓	
2018	[126]	TenMF	✓	✓				✓		✓						✓		
2018	[43]	GeoEISo	✓	✓						✓	✓					✓		
2018	[114]	GeoIE	✓							✓	✓						✓	
2019	[129]	MEAP-T					✓	✓		✓	✓						✓	
2019	[45]	MLR	✓	✓					✓		✓				✓	✓		
2019	[107]	APRA-SA	✓					✓			✓			✓		✓		
2019	[96]	STA	✓				✓	✓	✓	✓							✓	
2019	[143]	STGN	✓				✓	✓				✓					✓	
2020	[120]	HI-LDA	✓	✓		✓				✓	✓					✓		
2020	[117]	GAIMC	✓							✓	✓					✓		
2020	[142]	SPR	✓			✓				✓						✓		
2020	[57]	MMBE	✓	✓	✓		✓	✓		✓	✓	✓					✓	
2020	[118]	TECF	✓					✓	✓		✓	✓	✓	✓		✓		
Most Representatives			38	14	11	5	13	18	9	27	26	8	6	10	1	21	20	1
Total			218	116	108	42	73	134	90	141	139	66	44	98	19	152	118	14

by performing a preliminary analysis of the LBSNs data. On the other hand, social information is also widely modeled, partly because many of the datasets used, such as Gowalla, also provide the links of friendship between users. However, while there is more or less consensus on the importance of geographical information, this is not so clear for social information, where some researchers claim it is not so important [26, 42] while others state it plays an important role [28, 43]. One possible explanation for this effect is that even though users may share their tastes with friends (from the same or different cities), they may not visit the same POIs, in part because it is common

Table 5. Evaluation details of analyzed POI recommendation approaches sorted by publication year.

Year	Reference	Acronym	Filter data	Validation	Error	Ranking	Region Split	Check-in(✓) or POI(✗) split	Cold Start Analysis	Classic Non Personalized	Classic Personalized	Geographical	Random	Temporal	Other	System	Per User
2011	[127]	USG				✓		✗	✓		✓	✓	✓				✓
2012	[64]	LARS					✓	✗			✓	✓	✓			✓	
2012	[7]	(N.A.)	✓			✓	✓	✗			✓				✓		
2012	[131]	UPOI-Mine			✓	✓		✗			✓	✓					
2012	[92]	RW, Weighted-RW				✓	✓	✓		✓	✓	✓		✓		✓	
2013	[79]	(N.A.)				✓	✓	✓			✓	✓		✓			✓
2013	[74]	GT-BNMF				✓	✓	✓			✓		✓			✓	
2013	[27]	FPMC-LR	✓			✓		✓			✓			✓		✓	
2013	[40]	LRT	✓			✓		✗			✓		✓				✓
2013	[134]	UTE+SE	✓	✓		✓	✓	✗			✓	✓	✓				✓
2014	[130]	UPOI-Walk			✓	✓	✓	?			✓	✓					
2014	[135]	GTAG	✓	✓		✓		✗			✓	✓	✓				✓
2014	[73]	GeoMF	✓			✓		✗			✓		✓				✓
2014	[82]	IRenMF		✓		✓		✗			✓	✓	✓				✓
2014	[137]	LORE				✓		✓				✓		✓		✓	
2015	[77]	Poisson Geo-PFM	✓			✓	✓	✓			✓	✓	✓			✓	
2015	[68]	RankGeoFM		✓		✓	✓	✓			✓	✓		✓			✓
2015	[136]	GeoSoCa				✓	✓	✓				✓		✓		✓	
2015	[38]	PRME-G	✓	✓		✓	✓	✓		✓	✓			✓		✓	
2015	[41]	CAPRF	✓			✓	✓	✗			✓		✓				✓
2016	[119]	GE		✓		✓	✓	✓	✓			✓		✓			✓
2016	[65]	ASMF	✓			✓		✗	✓		✓	✓		✓			✓
2016	[146]	STELLAR	✓			✓		✓			✓			✓		✓	
2016	[52]	(N.A.)	✓			✓	✓	?			✓			✓		✓	
2016	[80]	WWO	✓			✓		✓						✓		✓	
2017	[145]	Geo-Teaser	✓			✓		✓			✓	✓		✓			✓
2017	[121]	PACE	✓			✓		✓				✓		✓			✓
2017	[99]	TGSC-PMF	✓		✓	✓		✓				✓	✓			✓	
2017	[51]	LBPR				✓	✓	✓			✓	✓		✓			✓
2017	[116]	VPOI	✓			✓	✓	✗	✓		✓		✓				✓
2018	[84]	SAE-NAD	✓			✓	✓	✗			✓	✓	✓				✓
2018	[85]	CARA	✓			✓		✓	✓		✓	✓		✓			✓
2018	[126]	TenMF	✓			✓		✗			✓	✓	✓				✓
2018	[43]	GeoEISo	✓			✓		✓			✓	✓	✓			✓	
2018	[114]	GeoIE	✓	✓		✓		✓				✓		✓			✓
2019	[129]	MEAP-T	✓	✓		✓	✓	✓		✓	✓			✓			✓
2019	[45]	MLR				✓	✓	✗			✓	✓	✓			✓	
2019	[107]	APRA-SA			✓	✓		✓			✓	✓	✓				✓
2019	[96]	STA		✓		✓		✓	✓			✓		✓			✓
2019	[143]	STGN	✓			✓	✓	✓	✓			✓		✓			✓
2020	[120]	HI-LDA				✓	✓	✓				✓	✓				✓	
2020	[117]	GAIMC	✓			✓	✓	✓				✓	✓			✓	
2020	[142]	SPR			✓	✓		✓			✓	✓	✓			✓	
2020	[57]	MMBE	✓			✓		✓				✓		✓		✓	
2020	[118]	TECF	✓			✓	✓	✗			✓	✓	✓				✓
	Most Representatives		27	9	5	43	28	C:27 P:16	7	3	32	33	21	20	1	17	24
	Total		171	53	25	294	144	C:205 P:71	38	35	173	198	152	118	14	135	135

for users to visit the locations closest to their centers of activity (home and work, basically) and most likely they will be different from their friends', even if they are "similar" in terms of tastes.

Textual or content information is also exploited by many approaches, especially those using some kind of probabilistic model such as topic modeling or the POI categories. This is because the features of the items (categories) in this domain are very distinctive and may even discriminate between different types of users in a LBSN; for example, a tourist may prefer to visit museums and

restaurants, whereas a local may prefer a bar or a shopping center. Finally, regarding temporal and sequential information, we observe that the latter is not so exploited (although some deep learning techniques make use of sequential information implicitly), but temporal information is taken into account in many approaches regardless of the technique used by the model under analysis, probably because of its flexibility to be introduced in almost any recommendation technique (usually at the cost of increased sparsity). The same trend can be found in Figure 3, where all the selected papers in the review, not only the representative ones, are shown in a year basis.

Now, when we analyze the type of model, a change in trend can be seen in the years between 2011 and 2015 and subsequent years, since for the former, proposals that used some type of collaborative system based on neighbors were reasonably popular, but not anymore. However, in subsequent years there has been a greater dominance of probabilistic and factorization proposals. This is something that already happened in traditional recommendation, where since the Netflix Prize in 2009 [12] in which matrix factorization models outperformed other traditional approaches, they have received more attention from the RS research community. In the same way, deep learning techniques have experienced a significant growth since 2017 in the POI recommendation area. This becomes evident in Figure 5 where, again, all the papers are included. Here, we observe that until 2017 there are less than 5 DL techniques included in the selected papers, but this type of model increases steadily year after year and in 2020 is the most extended approach. Finally, with respect to graph/link-based and hybrid models we observe that, in general, graph models are not widely used, whereas hybrid techniques, even though they are not widely used, they have been used throughout all the years collected in our analysis. One reasonable assumption for this is that hybrid proposals allow several elements to be combined into one, thus alleviating the possible drawbacks that each of them may show separately.

In the next section, we analyze in detail the evaluation aspects, including the split types that appear in the already discussed Table 4.

5 SYSTEMATIC REVIEW OF STATE-OF-THE-ART EVALUATION METHODOLOGIES

In this section, we continue the analysis on the state-of-the-art in POI recommendation presented previously but focusing on the evaluation aspects. Thus, we analyze the last columns of Table 4, together with Figure 4, which shows the characterization of some evaluation protocols, as shown in Section 3.4. We observe they are well distributed between random and temporal splits, although it is interesting to note that until 2014 the random partition predominated over the temporal split. However, from that year onwards the use of temporal splitting has increased steadily.

Nonetheless, there is still no common evaluation protocol to evaluate the performance of POI recommenders; this is an interesting but also a concerning conclusion, since this means that we might be comparing models that try to solve the same problem (POI recommendation) but, at the same time, we are evaluating them in very different ways, which in turn affects the performance of the algorithms. In particular, we have found surprising combinations regarding this, such as works with models using temporal information in their formulation that were running a random evaluation protocol in their evaluation, like [40, 134, 135, 146].

In any case, it should be noted that even if the entire community moves to a common splitting method, there are other aspects of the experimental settings that could affect the final performance of the algorithms and, hence, the validity of the published results. For example, how the candidate items to be ranked are selected is a well-known source for bias in RS evaluation [13], and it is not obvious how to compare results when all the items in the system are scored and ranked against strategies where only one item from the test set is ranked together with a random set of POIs, as it has become recently popular among deep learning techniques such as [85].

Because of this, in Table 5 we extend the evaluation aspects to be considered for the same works presented before. We now include whether some kind of data filtering is performed (to avoid both users and POIs with very few interactions), if a validation subset is used, the type of metric (error or ranking) reported, if the split was used based on geographical information, and if repetitions were considered or not (i.e., if the split was done by check-ins or POIs), as discussed in Section 3.4. We also decided to include if the authors performed some kind of cold-start analysis and the types of baselines considered in the experiments: whether they use classic non-customized recommendation baselines (popularity and/or random), classic and personalized baselines (user/item-based, BPR, or MFs), and geographic baselines (any algorithm that uses a geographical component).

Based on this information, we first observe that a relatively large number of articles apply some kind of filter in the data, the most typical one being to remove users or POIs with less than n interactions. It is important to note that we only put the mark ✓ if the authors specifically state in their paper that they filter the data, so there might be other proposals that use a pre-filtered dataset that do not count in the table; hence, these numbers are probably underestimating this aspect. Nonetheless, it is true that in some situations it might be necessary to make some pre-filtering of the data, but we must be careful since, if the filtering is too strict, we may end up evaluating the system with very little data, making the obtained results not generalizable. On the other hand, sometimes, instead of performing a simple training and test splits, researchers obtain a third subset of data to tune the parameters (called validation but different from a k-fold cross-validation). However, as we observe in the table, it is not very common in POI recommendation (as occurs in traditional recommendation). With respect to the type of metrics reported, there seems to be more consensus, since the vast majority of papers use some kind of ranking-based metric. Besides, most approaches that evaluate using rating prediction also use ranking evaluation, although there are very few approaches that only use rating prediction, like [123].

Regarding the region split column, we believe it is quite important when comparing research works in this domain, since algorithms executed in a worldwide dataset are not comparable against those executed in independent cities, mostly because the geographical influence is indeed affecting the recommenders in a very different way (and the obvious correlation between past and future user actions based on this dimension), depending on the type of split we are using. Similarly, depending on whether the split is done by check-ins or by POIs, it might affect the obtained results. Although this distinction might be subtle, if we analyze this aspect we observe there is a lot of disparity in the works, not leading to any clear conclusion. Let us consider for example that we select for each user 80% of their check-ins to train and the rest 20% to test, as we mentioned before, on many LBSNs there may be repetitions so the test set may be composed by check-ins that appear in the training set; however, if the split is made by POIs, we make sure to remove such repetitions and therefore we would not be recommending POIs that the user has already visited. At the same time, even though datasets in this domain are very sparse, few researchers perform a specific analysis on cold start, as evidenced from the values shown in the table (we denote as cold start those works that explicitly consider users or POIs with very few interactions, e.g., less than 5).

Now, in terms of the baselines used, although most of the approaches compare against baselines that can be categorized as classic algorithms such as MFs or k-NN, and many others use geographical influence, it is surprising that there are a limited number of works that test their approaches against very basic baselines like popularity, when it has been shown to be quite effective in domains with a high sparsity [13].

Finally, in Table 6 we show the articles processed according to the type of evaluation methodology used to evaluate their models (offline, online, and user studies). As we observe in this table, the vast majority of the researchers use an offline evaluation methodology. We argue this might be due to how expensive running such experiments become, in particular for those authors who

are already exploiting check-ins from LBSNs, so in most cases they do not need (even though it would provide complementary evidence) to check the user feedback directly. We would like to highlight, however, that even though we have not found any work in which the proposed models are evaluated in an online environment, we have found some proposals in which the authors claim that their algorithm is an online POI recommender, even though it is only evaluated in an offline scenario. Some examples of this type of works include: [7, 64, 96, 113, 128, 130]. Although this may be surprising, we are not the only researchers to notice that there are almost no online models in POI recommendation, at least using data from LBSNs [144].

6 SYSTEMATIC REVIEW OF DATASETS USED IN STATE-OF-THE-ART

There are several LBSNs that researchers use to explore the problem of POI recommendation and related tasks, but among all of them, there are four that stand out: Foursquare[8], Gowalla[9], Brightkite[10], and Yelp[11], as it is evidenced by Table 7, that shows the number of papers reporting data from each LBSN. Hence, researchers obtained data from these systems and used them for their experiments, even though the same data could be used for different purposes – i.e., not only the pure POI recommendation task we address here, but also for social or review recommendation.

Besides the differences in the actual recommendation task, which might be more or less obvious when comparing two research works, we noticed remarkable differences in the statistics reported for datasets that (in principle) belong to the same LBSN. The reason might be obvious: the datasets are obtained and pre-processed differently, however, since there is no *canonical name* for the datasets (as it occurs in other domains, e.g., with the MovieLens or Lastfm datasets), they are indistinctively referred as the name of the corresponding LBSN, which confuse the reader and other researchers into thinking that the same data is used in two works. We would also like to mention that in some works we have found strange statistics that we believe to be inaccurate. For example, in [21], the authors claim to use a Foursquare dataset but the same statistics can be found in [32, 47] for a Gowalla dataset. The statistics reported in [98, 113] are also strange as they report more users than check-ins (in this case, for a Foursquare dataset). To shed some light on this aspect, we now present some details about the most important datasets based on these LBSNs and later analyze some of these differences.

- Foursquare: it is possibly the most famous LBSN, which agrees with our statistics (see Table 7) that show it is the LBSN most frequently used by researchers, among the works included in our analysis. Users in Foursquare can visit a place, mark it as visited in the system (by *checking-in* in the venue) so their followers or friends could track it, like a venue, comment on it (by writing *tips*), and obtain recommendations from the system (since 2014 most of this functionality was derived to Swarm). In general, these check-ins cannot be obtained directly neither from its website nor its API; because of that, most researchers rely on other social networks where users share their interactions with Foursquare, mostly Twitter. Even though we will show different datasets from this LBSN in Tables 8 and 9, we consider important to emphasize that many papers that report using Foursquare include a url[12] that does not work anymore; these include the original work [42] and many more, such as [40, 54, 109, 137, 138].
- Gowalla: a LBSN that was acquired by Facebook in 2011. Most papers use the Gowalla dataset that can be found in the SNAP repository[13], such as [25, 48, 115]. As it also happens with

---

[8]https://foursquare.com
[9]It does not exist anymore since 2012.
[10]It was acquired by another social network in 2009 and does not exist anymore since 2012.
[11]https://www.yelp.com
[12]http://www.public.asu.edu/~hgao16/dataset.html
[13]http://snap.stanford.edu/data/loc-gowalla.html

Table 6. Evaluation methodologies used by papers included in our review.

Number of Papers	Evaluation methodology		
	Offline	Online	User study
Most Representatives	45	0	0
Total	306	0	5

Table 7. Papers included in our review that use a dataset from each LBSN.

Number of Papers	LBSN				
	Gowalla	Foursquare	Yelp	Brightkite	Other
Most Representatives	30	34	4	6	8
Total	156	199	54	40	43

other LBSNs, some researchers claim they use Gowalla, but they fail to provide any source to obtain such dataset [66, 92, 131, 148].

- Brightkite: a less popular LBSN, but used in a large number of research works because of its availability. In the same way as Gowalla, a dataset from this LBSN is included in the SNAP repository[14], which makes it easy to be used by researchers, since it is not available since 2012; some examples include [25, 54, 125].

- Yelp: this is a LBSN that focuses on businesses, rather than generic POIs like other LBSNs. It also differs from the other LBSNs in that users provide a rating based on 5 stars to the different businesses they visit; besides, users can also write a review about them. The Yelp dataset is available on its website[15] and can be obtained after agreeing on the dataset license; however, many papers refer to a different url[16] that does not work anymore, like [11, 50, 69, 121]; this is because this dataset was first released in the context of a challenge ran by Yelp, which has gone at least through 12 rounds where the data has been increased each time; this makes the comparisons even more difficult since it is not possible to get the dataset corresponding to a specific round, and this information is usually omitted in the papers.

- Others: in addition to the aforementioned LBSNs, some proposals work with datasets extracted from other systems, such as Jiepang (a Chinese LBSN similar to Foursquare) used in [70, 72, 73, 98], Weeplaces used in [9, 10], GeoLife used in [1, 149], and others less popular in our context, like Twitter and TripAdvisor.

While doing our systematic review, we found several versions of datasets coming from the same LBSN. For the sake of space and clarity, we show in Table 8 the LBSNs used by the research work with more citations (according to Scopus) for each year, together with some statistics of the dataset and other evaluation details reported in the experiments, such as the type of split and the evaluation metrics. Based on this information, we observe that all of them evaluate based on some notion of ranking quality; while it is true that this evidences researchers are taking into account the guidelines provided in the recommendation field as a whole [87], a possible reason for this is that the data to be predicted is not ratings anymore, but binary feedback: whether the user visited the POI or not. In any case, this table emphasizes an even more important problem: most researchers are only focused on accuracy, disregarding additional dimensions such as novelty, diversity, or serendipity that are becoming prevalent in recent years in the evaluation of RSs [22].

Table 8 also shows an interesting paradigm shift: those works prior to 2015 used a random split, and those more recent used a temporal split (except in 2020). We consider this a decisive signal, since it indicates that (at least for the works that are later more cited by colleagues) a more realistic type of split is being used, which would indeed make the proposed approaches easier to put in

---

[14]http://snap.stanford.edu/data/loc-brightkite.html

[15]https://www.yelp.com/dataset

[16]https://www.yelp.com/dataset_challenge

Table 8. Details of the experimental settings for the work with most citations each year (note each work appears once for each reported dataset). N.A. denotes that value is not provided in the paper. The columns Ref. and Cit. denote the original reference for that work and the number of citations, as of April 2021.

Details				Statistics				Evaluation Details	
Year	Ref.	Acronym	Cit.	Dataset	Users	POIs	Check-ins	Metrics used	Type of Split
2011	[127]	USG	784	Foursquare	153,577	96,229	N.A.	P, R	Random Per User
2011	[127]	USG	784	Whrrl	5,892	53,432	N.A.	P, R	Random Per User
2012	[7]	(N.A.)	477	Foursquare (NY)	2,886	N.A.	10,687	P, R	Other
2012	[7]	(N.A.)	477	Foursquare (LA)	228	N.A.	9,836	P, R	Other
2013	[134]	UTE, SE, UTE+SE	535	Foursquare	2,321	5,596	194,108	P, R	Random Per User
2013	[134]	UTE, SE, UTE+SE	535	Gowalla	10,162	24,250	456,988	P, R	Random Per User
2014	[73]	GeoMF	377	Jiepang	276,450	574,095	N.A.	P, R	Random Per User
2015	[68]	RankGeoFM	220	Foursquare	2,321	5,596	194,108	P, R	Temporal Per User
2015	[68]	RankGeoFM	220	Gowalla	10,162	24,250	456,988	P, R	Temporal Per User
2016	[119]	GE	184	Foursquare	114,508	62,462	1,434,668	Accuracy	Temporal Per User
2016	[119]	GE	184	Gowalla	107,092	1,280,969	6,442,892	Accuracy	Temporal Per User
2017	[121]	PACE	153	Gowalla	18,737	32,510	1,278,274	P, R, NDCG, MAP	Temporal Per User
2017	[121]	PACE	153	Yelp	30,887	18,995	860,888	P, R, NDCG, MAP	Temporal Per User
2018	[114]	GeoIE	49	Foursquare	6,118	88,193	172,961	P, R	Temporal Per User
2018	[114]	GeoIE	49	Gowalla	1,624	3,585	115,890	P, R	Temporal Per User
2019	[143]	STGN	45	Foursquare (CA)	49,005	206,097	425,691	Accuracy, MAP	Temporal Per User
2019	[143]	STGN	45	Foursquare (SIN)	30,887	18,995	860,888	Accuracy, MAP	Temporal Per User
2019	[143]	STGN	45	Gowalla	18,737	32,510	1,278,274	Accuracy, MAP	Temporal Per User
2019	[143]	STGN	45	Brightkite	51,406	772,967	4,747,288	Accuracy, MAP	Temporal Per User
2020	[118]	TECF	14	Foursquare	2,321	5,596	194,108	P, R	Random Per User
2020	[118]	TECF	14	Foursquare	10,162	24,250	456,988	P, R	Random Per User

Table 9. Statistics of reported versions for the Foursquare dataset in works included in our review, sorted by number of check-ins.

Users	POIs	Check-ins	Papers using this dataset
114,508	62,462	1,434,668	5
11,326	182,968	1,385,223	9
10,766	10,695	1,336,278	4
24,941	28,593	1,196,248	8
2,293	61,858	573,703	3
7,642	28,484	512,523	4
4,163	121,142	483,813	7
1,083	38,333	227,428	5
2,321	5,596	194,108	18
2,551	13,474	124,933	4

context in a real scenario. It is also positive that most of these works (it is by no means the same in general) contrast their approaches against two data sources, which makes the results easier to generalize. On the other hand, what can be considered as a worrying sign is that there are not two articles sharing the number of check-ins or users, except [68, 134] and [118], but even in this case, each work performs a different data splitting; moreover, there are even cases where some of the statistics are not included (like the number of items or check-ins). This makes it almost impossible to compare two research works without implementing everything from scratch, hence hindering reproducibility and the advancement of the field [103].

We were also surprised that in most cases the source code of the proposed method is not provided. In particular, among the papers with more citations, only [84, 85, 121, 142, 145] redirect to a repository with source code. With respect to the rest of the analyzed papers (that is, out of the 310 works), only [30, 55, 67, 69, 71, 75, 84, 85, 97, 121, 132, 142, 145] provide a url to download the source code of their algorithm.

As a final analysis, we present in Table 9 different versions of datasets extracted from Foursquare, considering this is the most widely used LBSN in the articles included in our review. In this selection we show the datasets used by more than two articles, since there are works using other variations not reported here but, for the sake of space, we focused only on those reported a minimum number

of times among the papers considered in our analysis[17]. Nevertheless, it is remarkable to observe the large difference in the number of check-ins, ranging from 45k to 2M interactions; as a consequence, the experiments presented in the different works are probably not comparable at all – even if they belong to the same LBSN – since the inherent properties of the system are not preserved: for instance, in some cases we have more users than items whereas in other cases it is the other way around; it is also possible that the levels of sparsity change dramatically, together with the number of cities/regions included in each dataset. It is interesting to observe that most datasets are seldom used, and the few cases where the same dataset is used by many works, it is because they belong to the same authors; see the supporting materials for an explicit list of the references using these datasets.

7 FUTURE RESEARCH DIRECTIONS AND OPEN ISSUES

In this section, we present some open issues we have identified after performing an analysis on the state-of-the-art on POI recommendation based on LBSNs, after that we list some potential future research lines we believe are in line with parallel developments in the field of recommender systems.

7.1 Open issues and research challenges

Although several research efforts have been devoted to the problem of POI recommendation, it is still possible to find unresolved issues in the field, which opens up opportunities to improve the area as a whole, for instance, because they are more aligned with the necessities of the final users and, probably, with industry practitioners. By analyzing the current proposals in POI recommendation based on LBSNs, we have observed some important open issues that need to be addressed. In the following, we group them according to the three main systematic reviews we performed: models or algorithms, evaluation methodologies, and datasets.

7.1.1 Open issues. Regarding algorithms, matrix factorization and, more recently, Deep Learning are very popular approaches in POI recommendation when using data from LBSNs; however, it is often difficult to explain why the recommendations from these methods are made since they behave like a *black box* and this can be problematic in some domains, in particular in tourism. In addition, we have also observed that most researchers do not test their approaches against other classic recommendation algorithms like simple CF methods or non-personalized item popularity, comparing only with other POI recommendation approaches. Similarly, the sequential information, despite its relevance in this domain, is not usually exploited, which sometimes prompts incorrect or not realistic evaluation methodologies.

In fact, about evaluation methodologies we consider the comparisons between different algorithms must always be as transparent and as fair as possible in order to determine which proposals are superior to others. Therefore, although in the papers analyzed in this survey there seems to be consensus in evaluating the approaches using IR metrics like Precision or Recall, this is not the case about how to perform the splits, as there are both random and temporal partitions (each of them with different variations), even though the latter ones are the only strategies that could simulate real scenarios. At the same time, the sparsity of the datasets used, whether or not they have been pre-filtered, etc., also affects the performance of the models, which in particular may prevent from having research works that are comparable between each other.

---

[17]For instance, a widely used version called Global-scale dataset from Yang et al. presented in [122] is not included in this table because only one paper reported the exact statistics as the original paper (which is actually not considered because it does not perform POI recommendation), whereas other works take subsets of it.

Regarding the datasets, although most proposals extract data from well-known LBSNs such as Foursquare, Gowalla, or Yelp, these datasets are often not comparable among them due to different decisions considered when filtering users or items, or even how the data were captured, which produces a large number of versions for each LBSN. Comparing datasets is even more difficult when some researchers do not provide complete statistics about the actual dataset used in the experiments, leading to data with completely different characteristics and inherent properties (sparsity, granularity of temporal, geographical, and social information, POI attributes, and so on) even when they belong to the same LBSN.

7.1.2 Discussion. As we have seen along the survey, the problem of POI recommendation is attractive to a growing number of researchers in the area of recommender systems. However, it may seem as if most of these issues have something in common: they make both the reproducibility and the generality of the proposed algorithms very difficult. Thus, in order to advance towards better systems and foster high-quality research, we recommend to:

- Explain in detail how the algorithms have been evaluated indicating the metrics used, the type of split and the rest of the models that have been used as baselines. In this regard, we suggest to test the proposed algorithm against specific POI recommendation models while also analyzing its performance against other baselines used in classical recommendation, such as neighbor-based algorithms, matrix factorization approaches, and the most-popular method. The evaluation methodology must be the same for all the algorithms and if it is necessary to make different experiments for choosing the parameters, this needs to be done for all the algorithms involved in the experiments and, if possible, with a validation subset independent of the test set.
- Clearly indicate the statistics of the used datasets, stating if any pre-processing step has been performed and showing the final details of the used data, including the number of users, POIs, and check-ins. This would help to detect the percentage of data that was removed to critically analyze if the filtered dataset is actually representative of the original dataset. We also strongly recommend researchers to use more than one dataset or, at least, to use different types of splits or more than one split from the same data if enough information is available.
- Finally, the easiest way to replicate a research work is by providing the code with a detailed description to achieve the same results mentioned in the paper; if this is not possible, the next best option is to, at least, provide the final datasets with which the algorithms were evaluated, so anyone interested in replicating it should not worry about that step of the evaluation pipeline.

In general, these recommendations aim to fix a lack of reproducible experimental settings that could hinder whether there is a significant improvement in the field, as already discussed in the RS and IR communities [3, 103].

7.2 Future directions

When preparing this survey, we have identified a number of future research directions, among them are the following.

7.2.1 Towards realistic methodologies. How realistic the evaluation methodologies or recommendation algorithms proposed really are? From our peespective, it may seem that sometimes the community is trying to solve a problem that will never arise in the real world, or at least, not in the terms it is being evaluated: using a global or worldwide test set (i.e., not divided by cities or countries) is not realistic, since a user at each point in time is only in one place and only interested in its surroundings. Because of this, any recommendation algorithm that exploits the geographical

information might be artificially benefited from this, but this can also be achieved simply by filtering the venues to be ranked at the evaluation step (as done in some works like [76]) instead of when modeling the problem.

Hence, we consider that researchers should formalize and critically think which task they want to solve and whether the evaluation methodology matches the task or if they are making it trivially easy or too much difficult *by design*. Related examples on this line include presenting a cold-start analysis while filtering users or items with too many interactions (see [26]) or not controling for new venues in test in a user basis (since users tend to visit the same venues they did in the past, so a simple baseline like returning the past profile of the user should be considered in the experiments).

7.2.2 Consider user types or roles. As recent studies show, check-in data can be used to characterize at least four types of travellers [35]: vacationers, explorers, voyagers, and globetrotters, thus going beyond the classical tourist roles that are usually considered (either leisure or business). In the future we expect specialized algorithms would be developed towards each of these user types, since they show different inherent needs and interests, as evidenced by their behavior when visiting POIs in a city, but also because of their personality traits. Moreover, other roles could be distinguished, even depending on the actual LBSN, since some systems might implicitly appeal to the social side of users, whereas others might be more attractive to POI owners, for instance; because of this, great care must be taken to transfer research results to a different LBSN if their underneath assumptions or interaction philosophies are not compatible.

7.2.3 Adversarial analysis and data quality. We have found very few papers where the quality of data used in POI recommendation is discussed, one example is [94][18]. Whereas in the classical recommendation problem the issue of robust recommendation – in the sense that the recommendation algorithm should not be too sensitive to attacks from malicious users – has been researched in the past and revisited recently with a different name [16, 34], there are several open issues about this topic regarding POI recommenders and LBSN data, such as: how can these types of systems be attacked? Is it possible to assess if the data already collected has suffered from such attacks? How can we detect and mitigate this malicious content?

It is interesting, however, that in some papers the authors remove some bogus interactions [93], as an indication that there is information that is better filtered out than left in the model, since these data points might influence the results. Nonetheless, a careful, detailed analysis of the impact of these points and how to detect them is still missing.

7.2.4 Novel information sources, biases, and privacy. As surveyed in Section 3.2, the POI recommendation problem typically considers several information sources, however we believe even more information sources will be available in the future, and some of them are ubiquitous at the moment but remain unexplored in this domain. For example, it is surprising the lack of works dealing with venue schedules; we attribute this to the fact that they tend to be used by optimization approaches which are more common when solving different problems, such as tour recommendation. In any case, we find it strange that they are not exploited for this (more simple) scenario, probably because of the difficulty to obtain trusted and consistent data. Images, for example, about the POIs have been recently used to infer the preferences of users. Another paradigmatic example is the Internet-of-Things and all the sensors (such as beacons) that are increasingly common in cities and touristic venues. While specific approaches tailored for a small set of sensor-ready POIs are starting to emerge, general architectures or frameworks aimed at solving the problem at a more global scale, or even interacting with POIs with and without sensors is, to the best of our knowledge, not investigated at the moment, despite its obvious interest and potential to attract users.

---

[18]This paper is not included in our analysis because it does not satisfy the requirements described in Section 1.2.

However, whenever more user information is exploited, concerns about biases, privacy, and ethical issues should be considered. While parts of these problems have been addressed in the past for LBSN data, we believe it should be revisited according to the novel information sources that might be available, and also because new recommendation approaches may entail or generate different statistical or cultural biases; moreover, recent approaches such as differential privacy aiming to provide valuable personalized recommendations while withholding sensitive information from the users could be an interesting solution for this type of systems [124].

7.2.5 Counterfactuals and translation into the real world. How does all this research translate to the real world with respect to the data? As we presented in Section 6, most datasets used in the experimental works come from a limited number of LBSNs (mostly, Foursquare), which may indicate a bias towards the requirements and needs from those systems. For instance, most works do not gather this information directly from Foursquare but through Twitter [122], evidencing the limitations of the collected data, which are probably incomplete and not uniform across the population of the LBSN under study.

Moreover, as it is common in other problems in the RS domain, the information available only refers to what the user actually did, not all the options presented by the system nor the discarded alternatives. In particular, this means that no negative information can be inferred, since only positive information (whenever there is an interaction between users and items) is recorded. Once this type of information would be available, the computation of counterfactuals and definition of intervention policies would allow to better align offline experimentation with online results [60].

7.2.6 Adding constraints. Constraint-based recommender systems are a family of recommendation approaches that are not among the most popular ones, because they have been applied in very few cases, since a deep knowledge of the domain is typically required [37]. However, we consider they may fit the POI recommendation problem since, often, the users face the RS with several restrictions or constraints: desired price of attractions, must-see venues, maximum length of the trip, and so on. Moreover, under special circumstances – such as an emergency situation or an unexpected crisis –, these constraints may be dictated by the venue owners or even the regional or national authorities; thus, it may become mandatory to satisfy such requirements. In this context, we foresee novel approaches that allow POI recommendation algorithms to incorporate constraints and adapt their suggestions to these varying conditions, perhaps by exploiting optimization techniques used for the tourist trip design problem [44].

7.2.7 Scalability and efficiency. Last but not least, one major drawback of most of the approaches surveyed in this review is that they exhibit expensive computational costs. This is because considering additional information dimensions, beyond the user-item interaction matrix, needs memory resources but also complex algorithms that are difficult to scale and execute efficiently. Therefore, a promising research direction would consist on defining approximated versions of well-known algorithms that could manage large amounts of multi-dimensional data, such as geographical, social, content, and user-item interactions – the most common information sources.

ACKNOWLEDGMENTS

REFERENCES

[1] Hamidu Abdel-Fatao, Jiuyong Li, and Jixue Liu. 2015. Unifying Spatial, Temporal and Semantic Features for an Effective GPS Trajectory-Based Location Recommendation. In *ADC (Lecture Notes in Computer Science)*, Vol. 9093. Springer, 41–53.

[2] Gediminas Adomavicius and Alexander Tuzhilin. 2005. Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. *IEEE Trans. Knowl. Data Eng.* 17, 6 (2005), 734–749.

[3] Timothy G. Armstrong, Alistair Moffat, William Webber, and Justin Zobel. 2009. Improvements that don't add up: ad-hoc retrieval results since 1998. In *CIKM*. ACM, 601–610.

[4] Gagangeet Singh Aujla, Anish Jindal, Rajat Chaudhary, Neeraj Kumar, Sahil Vashist, Neeraj Sharma, and Mohammad S. Obaidat. 2019. DLRS: Deep Learning-Based Recommender System for Smart Healthcare Ecosystem. In *ICC*. IEEE, 1–6.

[5] Ricardo A. Baeza-Yates and Berthier A. Ribeiro-Neto. 2011. *Modern Information Retrieval - the concepts and technology behind search, Second edition.* Pearson Education Ltd., Harlow, England.

[6] Marko Balabanovic and Yoav Shoham. 1997. Content-Based, Collaborative Recommendation. *Commun. ACM* 40, 3 (1997), 66–72.

[7] Jie Bao, Yu Zheng, and Mohamed F. Mokbel. 2012. Location-based and preference-aware recommendation using sparse geo-social networking data. In *SIGSPATIAL/GIS*. ACM, 199–208.

[8] Jie Bao, Yu Zheng, David Wilkie, and Mohamed F. Mokbel. 2015. Recommendations in location-based social networks: a survey. *GeoInformatica* 19, 3 (2015), 525–565.

[9] Ramesh Baral and Tao Li. 2016. MAPS: A Multi Aspect Personalized POI Recommender System. In *RecSys*. ACM, 281–284.

[10] Ramesh Baral, Dingding Wang, Tao Li, and Shu-Ching Chen. 2016. GeoTeCS: Exploiting Geographical, Temporal, Categorical and Social Aspects for Personalized POI Recommendation (Invited Paper). In *IRI*. IEEE Computer Society, 94–101.

[11] Ramesh Baral, Xiaolong Zhu, S. S. Iyengar, and Tao Li. 2018. ReEL: R eview Aware Explanation of Location Recommendation. In *UMAP*. ACM, 23–32.

[12] Robert M. Bell and Yehuda Koren. 2007. Lessons from the Netflix prize challenge. *SIGKDD Explorations* 9, 2 (2007), 75–79.

[13] Alejandro Bellogín, Pablo Castells, and Iván Cantador. 2017. Statistical biases in Information Retrieval metrics for recommender systems. *Inf. Retr. Journal* 20, 6 (2017), 606–634.

[14] Lucas Bernardi, Pablo Estevez, Matias Eidis, and Eqbal Osama. 2020. Recommending Accommodation Filters with Online Learning. In *ORSUM@RecSys (CEUR Workshop Proceedings)*, Vol. 2715. CEUR-WS.org.

[15] Cécile Bothorel, Neal Lathia, Romain Picot-Clémente, and Anastasios Noulas. 2018. Location Recommendation with Social Media Data. In *Social Information Access*. Lecture Notes in Computer Science, Vol. 10100. Springer, 624–653.

[16] Robin Burke, Michael P. O'Mahony, and Neil J. Hurley. 2015. Robust Collaborative Recommendation. In *Recommender Systems Handbook*. Springer, 961–995.

[17] Robin D. Burke. 2007. Hybrid Web Recommender Systems. In *The Adaptive Web (Lecture Notes in Computer Science)*, Vol. 4321. Springer, 377–408.

[18] HongYun Cai, Vincent W. Zheng, and Kevin Chen-Chuan Chang. 2018. A Comprehensive Survey of Graph Embedding: Problems, Techniques, and Applications. *IEEE Trans. Knowl. Data Eng.* 30, 9 (2018), 1616–1637.

[19] Pedro G. Campos, Fernando Díez, and Iván Cantador. 2014. Time-aware recommender systems: a comprehensive survey and analysis of existing evaluation protocols. *User Model. User-Adapt. Interact.* 24, 1-2 (2014), 67–119.

[20] Iván Cantador, Alejandro Bellogín, and David Vallet. 2010. Content-based recommendation in social tagging systems. In *RecSys*. ACM, 237–240.

[21] Keyan Cao, Jingjing Guo, Gongjie Meng, Haoli Liu, Yefan Liu, and Gui Li. 2020. Points-of-Interest Recommendation Algorithm Based on LBSN in Edge Computing Environment. *IEEE Access* 8 (2020), 47973–47983.

[22] Pablo Castells, Neil J. Hurley, and Saul Vargas. 2015. Novelty and Diversity in Recommender Systems. In *Recommender Systems Handbook*. Springer, 881–918.

[23] Dawei Chen, Cheng Soon Ong, and Lexing Xie. 2016. Learning Points and Routes to Recommend Trajectories. In *CIKM*. ACM, 2227–2232.

[24] Xuefeng Chen, Yifeng Zeng, Gao Cong, Shengchao Qin, Yanping Xiang, and Yuanshun Dai. 2015. On Information Coverage for Location Category Based Point-of-Interest Recommendation. In *AAAI*. AAAI Press, 37–43.

[25] Yan Chen, Xin Li, Lin Li, Guiquan Liu, and Guangdong Xu. 2016. Modeling User Mobility via User Psychological and Geographical Behaviors Towards Point-of-Interest Recommendation. In *DASFAA (1) (Lecture Notes in Computer Science)*, Vol. 9642. Springer, 364–380.

[26] Chen Cheng, Haiqin Yang, Irwin King, and Michael R. Lyu. 2016. A Unified Point-of-Interest Recommendation Framework in Location-Based Social Networks. *ACM TIST* 8, 1 (2016), 10:1–10:21.

[27] Chen Cheng, Haiqin Yang, Michael R. Lyu, and Irwin King. 2013. Where You Like to Go Next: Successive Point-of-Interest Recommendation. In *IJCAI*. IJCAI/AAAI, 2605–2611.

[28] Nai-Hung Cheng and Chia-Hui Chang. 2013. Evaluation of Social, Geography, Location Effects for Point-of-Interest Recommendation. In *ICDM Workshops*. IEEE Computer Society, 766–772.

[29] Eunjoon Cho, Seth A. Myers, and Jure Leskovec. 2011. Friendship and mobility: user movement in location-based social networks. In *KDD*. ACM, 1082–1090.

[30] Giannis Christoforidis, Pavlos Kefalas, Apostolos Papadopoulos, and Yannis Manolopoulos. 2018. Recommendation of Points-of-Interest Using Graph Embeddings. In *DSAA*. IEEE, 31–40.

[31] Giannis Christoforidis, Pavlos Kefalas, Apostolos N. Papadopoulos, and Yannis Manolopoulos. 2019. Recommending Points of Interest in LBSNs Using Deep Learning Techniques. In *INISTA*. IEEE, 1–6.

[32] Song Chuang, Wen Junhao, and Li Shun. 2019. Personalized POI Recommendation Based on Check-in Data and Geographical-Regional Influence. In *Proceedings of the 3rd International Conference on Machine Learning and Soft Computing*. Association for Computing Machinery, 128–133.

[33] Marco de Gemmis, Pasquale Lops, Cataldo Musto, Fedelucio Narducci, and Giovanni Semeraro. 2015. Semantics-Aware Content-Based Recommender Systems. In *Recommender Systems Handbook*. Springer, 119–159.

[34] Yashar Deldjoo, Tommaso Di Noia, and Felice Antonio Merra. 2020. Adversarial Machine Learning in Recommender Systems (AML-RecSys). In *WSDM*. ACM, 869–872.

[35] Linus W. Dietz, Rinita Roy, and Wolfgang Wörndl. 2019. Characterisation of Traveller Types Using Check-In Data from Location-Based Social Networks. In *ENTER*. Springer, 15–26.

[36] Zhijun Ding, Xiaolun Li, Changjun Jiang, and Mengchu Zhou. 2018. Objectives and State-of-the-Art of Location-Based Social Network Recommender Systems. *ACM Comput. Surv.* 51, 1 (2018), 18:1–18:28.

[37] Alexander Felfernig, Gerhard Friedrich, Dietmar Jannach, and Markus Zanker. 2015. Constraint-Based Recommender Systems. In *Recommender Systems Handbook*. Springer, 161–190.

[38] 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 *IJCAI*. AAAI Press, 2069–2075.

[39] Md. Shafiul Alam Forhad, Mohammad Shamsul Arefin, A. S. M. Kayes, Khandakar Ahmed, Mohammad Jabed Morshed Chowdhury, and Indika Kumara. 2021. An Effective Hotel Recommendation System through Processing Heterogeneous Data. *Electronics* 10, 16 (2021).

[40] Huiji Gao, Jiliang Tang, Xia Hu, and Huan Liu. 2013. Exploring temporal effects for location recommendation on location-based social networks. In *RecSys*. ACM, 93–100.

[41] Huiji Gao, Jiliang Tang, Xia Hu, and Huan Liu. 2015. Content-Aware Point of Interest Recommendation on Location-Based Social Networks. In *AAAI*. AAAI Press, 1721–1727.

[42] Huiji Gao, Jiliang Tang, and Huan Liu. 2012. gSCorr: modeling geo-social correlations for new check-ins on location-based social networks. In *CIKM*. ACM, 1582–1586.

[43] Rong Gao, Jing Li, Xuefei Li, Chengfang Song, and Yifei Zhou. 2018. A personalized point-of-interest recommendation model via fusion of geo-social information. *Neurocomputing* 273 (2018), 159–170.

[44] Damianos Gavalas, Charalampos Konstantopoulos, Konstantinos Mastakas, and Grammati E. Pantziou. 2014. A survey on algorithmic approaches for solving tourist trip design problems. *J. Heuristics* 20, 3 (2014), 291–328.

[45] Bingrui Geng, Licheng Jiao, Maoguo Gong, Lingling Li, and Yue Wu. 2019. A two-step personalized location recommendation based on multi-objective immune algorithm. *Inf. Sci.* 475 (2019), 161–181.

[46] Asela Gunawardana and Guy Shani. 2015. Evaluating Recommender Systems. In *Recommender Systems Handbook*. Springer, 265–308.

[47] Lei Guo, Haoran Jiang, Xiyu Liu, and Changming Xing. 2019. Network Embedding-Aware Point-of-Interest Recommendation in Location-Based Social Networks. *Complexity* 2019 (2019), 3574194:1–3574194:18.

[48] Qing Guo, Yi Huang, and Yin-Leng Theng. 2015. Topic-Sensitive Location Recommendation with Spatial Awareness. In *WI-IAT (1)*. IEEE Computer Society, 237–243.

[49] Qing Guo, Zhu Sun, Jie Zhang, and Yin-Leng Theng. 2019. Modeling Heterogeneous Influences for Point-of-Interest Recommendation in Location-Based Social Networks. In *ICWE (Lecture Notes in Computer Science)*, Vol. 11496. Springer, 72–80.

[50] Saurabh Gupta, Sayan Pathak, and Bivas Mitra. 2015. Complementary Usage of Tips and Reviews for Location Recommendation in Yelp. In *PAKDD (2) (Lecture Notes in Computer Science)*, Vol. 9078. Springer, 720–731.

[51] Jing He, Xin Li, and Lejian Liao. 2017. Category-aware Next Point-of-Interest Recommendation via Listwise Bayesian Personalized Ranking. In *IJCAI*. ijcai.org, 1837–1843.

[52] Jing He, Xin Li, Lejian Liao, Dandan Song, and William K. Cheung. 2016. Inferring a Personalized Next Point-of-Interest Recommendation Model with Latent Behavior Patterns. In *AAAI*. AAAI Press, 137–143.

[53] Ruining He and Julian McAuley. 2016. Fusing Similarity Models with Markov Chains for Sparse Sequential Recommendation. In *ICDM*. IEEE, 191–200.

[54] Saeid Hosseini and Lei Thor Li. 2016. Point-Of-Interest Recommendation Using Temporal Orientations of Users and Locations. In *DASFAA (1) (Lecture Notes in Computer Science)*, Vol. 9642. Springer, 330–347.

[55] Bo Hu and Martin Ester. 2014. Social Topic Modeling for Point-of-Interest Recommendation in Location-Based Social Networks. In *ICDM*. IEEE Computer Society, 845–850.

[56] Haosheng Huang, Georg Gartner, Jukka Matthias Krisp, Martin Raubal, and Nico Van de Weghe. 2018. Location based services: ongoing evolution and research agenda. *J. Locat. Based Serv.* 12, 2 (2018), 63–93.

[57] Liwei Huang, Yutao Ma, Yanbo Liu, and Arun Kumar Sangaiah. 2020. Multi-modal Bayesian embedding for point-of-interest recommendation on location-based cyber-physical-social networks. *Future Gener. Comput. Syst.* 108 (2020), 1119–1128.

[58] Mozhgan Karimi, Dietmar Jannach, and Michael Jugovac. 2018. News recommender systems - Survey and roads ahead. *Inf. Process. Manage.* 54, 6 (2018), 1203–1227.

[59] Harmanjeet Kaur, Neeraj Kumar, and Shalini Batra. 2018. An efficient multi-party scheme for privacy preserving collaborative filtering for healthcare recommender system. *Future Gener. Comput. Syst.* 86 (2018), 297–307.

[60] Emre Kiciman and Amit Sharma. 2019. Causal Inference and Counterfactual Reasoning (3hr Tutorial). In *WSDM*. ACM, 828–829.

[61] Irena Koprinska and Kalina Yacef. 2015. People-to-People Reciprocal Recommenders. In *Recommender Systems Handbook*. Springer, 545–567.

[62] Yehuda Koren and Robert M. Bell. 2015. Advances in Collaborative Filtering. In *Recommender Systems Handbook*. Springer, 77–118.

[63] Takeshi Kurashima, Tomoharu Iwata, Go Irie, and Ko Fujimura. 2010. Travel route recommendation using geotags in photo sharing sites. In *CIKM*. ACM, 579–588.

[64] Justin J. Levandoski, Mohamed Sarwat, Ahmed Eldawy, and Mohamed F. Mokbel. 2012. LARS: A Location-Aware Recommender System. In *ICDE*. IEEE Computer Society, 450–461.

[65] Huayu Li, Yong Ge, Richang Hong, and Hengshu Zhu. 2016. Point-of-Interest Recommendations: Learning Potential Check-ins from Friends. In *KDD*. ACM, 975–984.

[66] Huayu Li, Yong Ge, Defu Lian, and Hao Liu. 2017. Learning User's Intrinsic and Extrinsic Interests for Point-of-Interest Recommendation: A Unified Approach. In *IJCAI*. ijcai.org, 2117–2123.

[67] Ranzhen Li, Yanyan Shen, and Yanmin Zhu. 2018. Next Point-of-Interest Recommendation with Temporal and Multi-level Context Attention. In *ICDM*. IEEE Computer Society, 1110–1115.

[68] Xutao Li, Gao Cong, Xiaoli Li, Tuan-Anh Nguyen Pham, and Shonali Krishnaswamy. 2015. Rank-GeoFM: A Ranking based Geographical Factorization Method for Point of Interest Recommendation. In *SIGIR*. ACM, 433–442.

[69] Xin Li, Guandong Xu, Enhong Chen, and Yu Zong. 2015. Learning recency based comparative choice towards point-of-interest recommendation. *Expert Syst. Appl.* 42, 9 (2015), 4274–4283.

[70] Defu Lian, Yong Ge, Fuzheng Zhang, Nicholas Jing Yuan, Xing Xie, Tao Zhou, and Yong Rui. 2015. Content-Aware Collaborative Filtering for Location Recommendation Based on Human Mobility Data. In *ICDM*. IEEE Computer Society, 261–270.

[71] Defu Lian, Yongji Wu, Yong Ge, Xing Xie, and Enhong Chen. 2020. Geography-Aware Sequential Location Recommendation. In *KDD*. ACM, 2009–2019.

[72] Defu Lian, Zhenyu Zhang, Yong Ge, Fuzheng Zhang, Nicholas Jing Yuan, and Xing Xie. 2016. Regularized Content-Aware Tensor Factorization Meets Temporal-Aware Location Recommendation. In *ICDM*. IEEE Computer Society, 1029–1034.

[73] Defu Lian, Cong Zhao, Xing Xie, Guangzhong Sun, Enhong Chen, and Yong Rui. 2014. GeoMF: joint geographical modeling and matrix factorization for point-of-interest recommendation. In *KDD*. ACM, 831–840.

[74] Bin Liu, Yanjie Fu, Zijun Yao, and Hui Xiong. 2013. Learning geographical preferences for point-of-interest recommendation. In *KDD*. ACM, 1043–1051.

[75] Bo Liu, Qing Meng, Hengyuan Zhang, Kun Xu, and Jiuxin Cao. 2020. VGMF: Visual contents and geographical influence enhanced point-of-interest recommendation in location-based social network. *Transactions on Emerging Telecommunications Technologies* n/a, n/a (2020), e3889. https://doi.org/10.1002/ett.3889 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/ett.3889

[76] Bin Liu and Hui Xiong. 2013. Point-of-Interest Recommendation in Location Based Social Networks with Topic and Location Awareness. In *SDM*. SIAM, 396–404.

[77] Bin Liu, Hui Xiong, Spiros Papadimitriou, Yanjie Fu, and Zijun Yao. 2015. A General Geographical Probabilistic Factor Model for Point of Interest Recommendation. *IEEE Trans. Knowl. Data Eng.* 27, 5 (2015), 1167–1179.

[78] Guimei Liu, Tam T. Nguyen, Gang Zhao, Wei Zha, Jianbo Yang, Jianneng Cao, Min Wu, Peilin Zhao, and Wei Chen. 2016. Repeat Buyer Prediction for E-Commerce. In *KDD*. ACM, 155–164.

[79] Xin Liu, Yong Liu, Karl Aberer, and Chunyan Miao. 2013. Personalized point-of-interest recommendation by mining users' preference transition. In *CIKM*. ACM, 733–738.

[80] Yanchi Liu, Chuanren Liu, Bin Liu, Meng Qu, and Hui Xiong. 2016. Unified Point-of-Interest Recommendation with Temporal Interval Assessment. In *KDD*. ACM, 1015–1024.

[81] Yiding Liu, Tuan-Anh Pham, Gao Cong, and Quan Yuan. 2017. An Experimental Evaluation of Point-of-interest Recommendation in Location-based Social Networks. *PVLDB* 10, 10 (2017), 1010–1021.

[82] Yong Liu, Wei Wei, Aixin Sun, and Chunyan Miao. 2014. Exploiting Geographical Neighborhood Characteristics for Location Recommendation. In *CIKM*. ACM, 739–748.

[83] Pasquale Lops, Dietmar Jannach, Cataldo Musto, Toine Bogers, and Marijn Koolen. 2019. Trends in content-based recommendation - Preface to the special issue on Recommender systems based on rich item descriptions. *User Model. User Adapt. Interact.* 29, 2 (2019), 239–249.

[84] Chen Ma, Yingxue Zhang, Qinglong Wang, and Xue Liu. 2018. Point-of-Interest Recommendation: Exploiting Self-Attentive Autoencoders with Neighbor-Aware Influence. In *CIKM*. ACM, 697–706.

[85] Jarana Manotumruksa, Craig Macdonald, and Iadh Ounis. 2018. A Contextual Attention Recurrent Architecture for Context-Aware Venue Recommendation. In *SIGIR*. ACM, 555–564.

[86] Pramit Mazumdar, Bidyut Kr. Patra, and Korra Sathya Babu. 2020. Cold-start Point-of-interest Recommendation through Crowdsourcing. *ACM Trans. Web* 14, 4 (2020), 19:1–19:36.

[87] Sean M. McNee, John Riedl, and Joseph A. Konstan. 2006. Being accurate is not enough: how accuracy metrics have hurt recommender systems. In *CHI Extended Abstracts*. ACM, 1097–1101.

[88] Harvey J. Miller. 2004. Tobler's First Law and Spatial Analysis. *Annals of the Association of American Geographers* 94, 2 (2004), 284–289. https://doi.org/10.1111/j.1467-8306.2004.09402005.x

[89] Marwa Naili, Anja Habacha Chaïbi, and Henda Hajjami Ben Ghezala. 2017. Comparative study of word embedding methods in topic segmentation. In *KES (Procedia Computer Science)*, Vol. 112. Elsevier, 340–349.

[90] Weizhi Nie, Anan Liu, Xiaorong Zhu, and Yuting Su. 2016. Quality models for venue recommendation in location-based social network. *Multimedia Tools Appl.* 75, 20 (2016), 12521–12534.

[91] Xia Ning, Christian Desrosiers, and George Karypis. 2015. A Comprehensive Survey of Neighborhood-Based Recommendation Methods. In *Recommender Systems Handbook*. Springer, 37–76.

[92] Anastasios Noulas, Salvatore Scellato, Neal Lathia, and Cecilia Mascolo. 2012. A Random Walk around the City: New Venue Recommendation in Location-Based Social Networks. In *SocialCom/PASSAT*. IEEE Computer Society, 144–153.

[93] Enrico Palumbo, Giuseppe Rizzo, Raphaël Troncy, and Elena Baralis. 2017. Predicting Your Next Stop-over from Location-based Social Network Data with Recurrent Neural Networks. In *RecTour@RecSys (CEUR Workshop Proceedings)*, Vol. 1906. CEUR-WS.org, 1–8.

[94] Evangelos E. Papalexakis, Konstantinos Pelechrinis, and Christos Faloutsos. 2014. Spotting misbehaviors in location-based social networks using tensors. In *WWW (Companion Volume)*. ACM, 551–552.

[95] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. DeepWalk: online learning of social representations. In *KDD*. ACM, 701–710.

[96] Tieyun Qian, Bei Liu, Quoc Viet Hung Nguyen, and Hongzhi Yin. 2019. Spatiotemporal Representation Learning for Translation-Based POI Recommendation. *ACM Trans. Inf. Syst.* 37, 2 (2019), 18:1–18:24.

[97] Hossein A. Rahmani, Mohammad Aliannejadi, Mitra Baratchi, and Fabio Crestani. 2020. Joint Geographical and Temporal Modeling Based on Matrix Factorization for Point-of-Interest Recommendation. In *ECIR (1) (Lecture Notes in Computer Science)*, Vol. 12035. Springer, 205–219.

[98] Logesh Ravi and V. Subramaniyaswamy. 2017. A Reliable Point of Interest Recommendation based on Trust Relevancy between Users. *Wireless Personal Communications* 97, 2 (2017), 2751–2780.

[99] Xingyi Ren, Meina Song, Haihong E, and Junde Song. 2017. Context-aware probabilistic matrix factorization modeling for point-of-interest recommendation. *Neurocomputing* 241 (2017), 38–55.

[100] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian Personalized Ranking from Implicit Feedback. In *UAI*. AUAI Press, 452–461.

[101] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2010. Factorizing personalized Markov chains for next-basket recommendation. In *WWW*. ACM, 811–820.

[102] Francesco Ricci, Lior Rokach, and Bracha Shapira. 2015. Recommender Systems: Introduction and Challenges. In *Recommender Systems Handbook*. Springer, 1–34.

[103] Alan Said and Alejandro Bellogín. 2014. Comparative recommender system evaluation: benchmarking recommendation frameworks. In *RecSys*. ACM, 129–136.

[104] Pablo Sánchez and Alejandro Bellogín. 2020. Applying reranking strategies to route recommendation using sequence-aware evaluation. *User Model. User Adapt. Interact.* 30, 4 (2020), 659–725.

[105] Guy Shani, David Heckerman, and Ronen I. Brafman. 2005. An MDP-Based Recommender System. *Journal of Machine Learning Research* 6 (2005), 1265–1295.

[106] Junge Shen, Cheng Deng, and Xinbo Gao. 2016. Attraction recommendation: Towards personalized tourism via collective intelligence. *Neurocomputing* 173 (2016), 789–798.

[107] Yali Si, Fuzhi Zhang, and Wenyuan Liu. 2019. An adaptive point-of-interest recommendation method for location-based social networks based on user activity and spatial features. *Knowl.-Based Syst.* 163 (2019), 267–282.

[108] Steffen Staab, Hannes Werthner, Francesco Ricci, Alexander Zipf, Ulrike Gretzel, Daniel R. Fesenmaier, Cécile Paris, and Craig A. Knoblock. 2002. Intelligent Systems for Tourism. *IEEE Intell. Syst.* 17, 6 (2002), 53–64.

[109] Torin Stepan, Jason M. Morawski, Scott Dick, and James Miller. 2016. Incorporating Spatial, Temporal, and Social Context in Recommendations for Location-Based Social Networks. *IEEE Trans. Comput. Social Systems* 3, 4 (2016), 164–175.

[110] Panagiotis Symeonidis, Dimitrios Ntempos, and Yannis Manolopoulos. 2014. *Recommender Systems for Location-based Social Networks.* Springer.

[111] Panagiotis Symeonidis, Alexis Papadimitriou, Yannis Manolopoulos, Pinar Senkul, and Ismail Hakki Toroslu. 2011. Geo-social recommendations based on incremental tensor reduction and local path traversal. In *GIS-LBSN*. ACM, 89–96.

[112] Christoph Trattner, Alexander Oberegger, Leandro Balby Marinho, and Denis Parra. 2018. Investigating the utility of the weather context for point of interest recommendations. *J. of IT & Tourism* 19, 1-4 (2018), 117–150.

[113] Henan Wang, Guoliang Li, and Jianhua Feng. 2014. Group-Based Personalized Location Recommendation on Social Networks. In *APWeb (Lecture Notes in Computer Science)*, Vol. 8709. Springer, 68–80.

[114] Hao Wang, Huawei Shen, Wentao Ouyang, and Xueqi Cheng. 2018. Exploiting POI-Specific Geographical Influence for Point-of-Interest Recommendation. In *IJCAI*. ijcai.org, 3877–3883.

[115] Hao Wang, Manolis Terrovitis, and Nikos Mamoulis. 2013. Location recommendation in location-based social networks using user check-in data. In *SIGSPATIAL/GIS*. ACM, 364–373.

[116] Suhang Wang, Yilin Wang, Jiliang Tang, Kai Shu, Suhas Ranganath, and Huan Liu. 2017. What Your Images Reveal: Exploiting Visual Contents for Point-of-Interest Recommendation. In *WWW*. ACM, 391–400.

[117] Wei Wang, Junyang Chen, Jinzhong Wang, Junxin Chen, and Zhiguo Gong. 2020. Geography-Aware Inductive Matrix Completion for Personalized Point-of-Interest Recommendation in Smart Cities. *IEEE Internet Things J.* 7, 5 (2020), 4361–4370.

[118] Wei Wang, Junyang Chen, Jinzhong Wang, Junxin Chen, Jinquan Liu, and Zhiguo Gong. 2020. Trust-Enhanced Collaborative Filtering for Personalized Point of Interests Recommendation. *IEEE Trans. Ind. Informatics* 16, 9 (2020), 6124–6132.

[119] Min Xie, Hongzhi Yin, Hao Wang, Fanjiang Xu, Weitong Chen, and Sen Wang. 2016. Learning Graph-based POI Embedding for Location-based Recommendation. In *CIKM*. ACM, 15–24.

[120] Xi Xiong, Shaojie Qiao, Nan Han, Fei Xiong, Zhan Bu, Rong-Hua Li, Kun Yue, and Guan Yuan. 2020. Where to go: An effective point-of-interest recommendation framework for heterogeneous social networks. *Neurocomputing* 373 (2020), 56–69.

[121] Carl Yang, Lanxiao Bai, Chao Zhang, Quan Yuan, and Jiawei Han. 2017. Bridging Collaborative Filtering and Semi-Supervised Learning: A Neural Approach for POI Recommendation. In *KDD*. ACM, 1245–1254.

[122] Dingqi Yang, Daqing Zhang, and Bingqing Qu. 2016. Participatory Cultural Mapping Based on Collective Behavior Data in Location-Based Social Networks. *ACM TIST* 7, 3 (2016), 30:1–30:23.

[123] Dingqi Yang, Daqing Zhang, Zhiyong Yu, and Zhu Wang. 2013. A sentiment-enhanced personalized location recommendation system. In *HT*. ACM, 119–128.

[124] Grace Hui Yang and Sicong Zhang. 2018. Differential Privacy for Information Retrieval. In *WSDM*. ACM, 777–778.

[125] Lina Yao, Quan Z. Sheng, Yongrui Qin, Xianzhi Wang, Ali Shemshadi, and Qi He. 2015. Context-aware Point-of-Interest Recommendation Using Tensor Factorization with Social Regularization. In *SIGIR*. ACM, 1007–1010.

[126] Lina Yao, Quan Z. Sheng, Xianzhi Wang, Wei Emma Zhang, and Yongrui Qin. 2018. Collaborative Location Recommendation by Integrating Multi-dimensional Contextual Information. *ACM Trans. Internet Techn.* 18, 3 (2018), 32:1–32:24.

[127] Mao Ye, Peifeng Yin, Wang-Chien Lee, and Dik Lun Lee. 2011. Exploiting geographical influence for collaborative point-of-interest recommendation. In *SIGIR*. ACM, 325–334.

[128] Hongzhi Yin, Bin Cui, Ling Chen, Zhiting Hu, and Chengqi Zhang. 2015. Modeling Location-Based User Rating Profiles for Personalized Recommendation. *TKDD* 9, 3 (2015), 19:1–19:41.

[129] Haochao Ying, Jian Wu, Guandong Xu, Yanchi Liu, Tingting Liang, Xiao Zhang, and Hui Xiong. 2019. Time-aware metric embedding with asymmetric projection for successive POI recommendation. *World Wide Web* 22, 5 (2019), 2209–2224.

[130] Josh Jia-Ching Ying, Wen-Ning Kuo, Vincent S. Tseng, and Eric Hsueh-Chan Lu. 2014. Mining User Check-In Behavior with a Random Walk for Urban Point-of-Interest Recommendations. *ACM TIST* 5, 3 (2014), 40:1–40:26.

[131] Josh Jia-Ching Ying, Eric Hsueh-Chan Lu, Wen-Ning Kuo, and Vincent S. Tseng. 2012. Urban point-of-interest recommendation by mining user check-in behaviors. In *UrbComp@KDD*. ACM, 63–70.

[132] Dongjin Yu, Kaihui Xu, and Dongjing Wang. 2019. Modeling User Contextual Behavior Semantics with Geographical Influence for Point-Of-Interest Recommendation. In *SEKE*. KSI Research Inc. and Knowledge Systems Institute Graduate School, 373–484.

[133] Yonghong Yu and Xingguo Chen. 2015. A survey of point-of-interest recommendation in location-based social networks. In *Workshops at the Twenty-Ninth AAAI Conference on Artificial Intelligence*.

[134] Quan Yuan, Gao Cong, Zongyang Ma, Aixin Sun, and Nadia Magnenat-Thalmann. 2013. Time-aware point-of-interest recommendation. In *SIGIR*. ACM, 363–372.

[135] Quan Yuan, Gao Cong, and Aixin Sun. 2014. Graph-based Point-of-interest Recommendation with Geographical and Temporal Influences. In *CIKM*. ACM, 659–668.

[136] Jia-Dong Zhang and Chi-Yin Chow. 2015. GeoSoCa: Exploiting Geographical, Social and Categorical Correlations for Point-of-Interest Recommendations. In *SIGIR*. ACM, 443–452.

[137] Jia-Dong Zhang, Chi-Yin Chow, and Yanhua Li. 2014. LORE: exploiting sequential influence for location recommendations. In *SIGSPATIAL/GIS*. ACM, 103–112.

[138] Jia-Dong Zhang, Chi-Yin Chow, and Yanhua Li. 2015. iGeoRec: A Personalized and Efficient Geographical Location Recommendation Framework. *IEEE Trans. Services Computing* 8, 5 (2015), 701–714.

[139] Jia-Dong Zhang, Chi-Yin Chow, and Yu Zheng. 2015. ORec: An Opinion-Based Point-of-Interest Recommendation Framework. In *CIKM*. ACM, 1641–1650.

[140] Shuai Zhang, Lina Yao, Aixin Sun, and Yi Tay. 2019. Deep Learning Based Recommender System: A Survey and New Perspectives. *ACM Comput. Surv.* 52, 1 (2019), 5:1–5:38.

[141] Wei Zhang and Jianyong Wang. 2015. Location and Time Aware Social Collaborative Retrieval for New Successive Point-of-Interest Recommendation. In *CIKM*. ACM, 1221–1230.

[142] Guoshuai Zhao, Peiliang Lou, Xueming Qian, and Xingsong Hou. 2020. Personalized location recommendation by fusing sentimental and spatial context. *Knowl. Based Syst.* 196 (2020), 105849.

[143] Pengpeng Zhao, Haifeng Zhu, Yanchi Liu, Jiajie Xu, Zhixu Li, Fuzhen Zhuang, Victor S. Sheng, and Xiaofang Zhou. 2019. Where to Go Next: A Spatio-Temporal Gated Network for Next POI Recommendation. In *AAAI*. AAAI Press, 5877–5884.

[144] Shenglin Zhao, Michael R. Lyu, and Irwin King. 2018. *Point-of-Interest Recommendation in Location-Based Social Networks*. Springer.

[145] Shenglin Zhao, Tong Zhao, Irwin King, and Michael R. Lyu. 2017. Geo-Teaser: Geo-Temporal Sequential Embedding Rank for Point-of-interest Recommendation. In *WWW (Companion Volume)*. ACM, 153–162.

[146] Shenglin Zhao, Tong Zhao, Haiqin Yang, Michael R. Lyu, and Irwin King. 2016. STELLAR: Spatial-Temporal Latent Ranking for Successive Point-of-Interest Recommendation. In *AAAI*. AAAI Press, 315–322.

[147] Xin Zheng, Jialong Han, and Aixin Sun. 2018. A Survey of Location Prediction on Twitter. *IEEE Trans. Knowl. Data Eng.* 30, 9 (2018), 1652–1671.

[148] Dequan Zhou and Xin Wang. 2014. Probabilistic Category-based Location Recommendation Utilizing Temporal Influence and Geographical Influence. In *DSAA*. IEEE, 115–121.

[149] Liang Zhu, Changqiao Xu, Jianfeng Guan, and Hongke Zhang. 2017. SEM-PPA: A semantical pattern and preference-aware service mining method for personalized point of interest recommendation. *J. Network and Computer Applications* 82 (2017), 35–46.