Towards Development of a Hypertext Induced Topic Search Based Point of Interest Recommender System for Location Based Social Networks

[1]John Vaseekaran S and [2]Srinivasan N
[1]Sathyabama Institute of Science and Technology, Chennai, Tamil Nadu, India.
[2]Department of Computer Science and Engineering, Rajalakshmi Engineering College, Chennai, Tamil Nadu, India.
[1]vaseekaranjohn168@gmail.com, [2]srinivasan.n@rajalakshmi.edu.in

Correspondence should be addressed to John Vaseekaran S : vaseekaranjohn168@gmail.com

Article Info
Journal of Machine and Computing (https://anapub.co.ke/journals/jmc/jmc.html)
Doi : https://doi.org/10.53759/7669/jmc202505017
Received 30 April 2024; Revised from 28 September 2024; Accepted 04 November 2024.
Available online 05 January 2025.

Abstract – Location-based social networks (LBSN) have a significant issue in the suggestion of points of interest (POIs) due to the sparsity of data, implicit input from users, and individual preferences. In most of the LBSN systems, there is no simple rating method for POIs, which is a major drawback for many users. Due to a lack of acceptable connections, such algorithms tend to provide a list of POIs that the user cannot consistently visit. There are many applications for the link data analysis, and the Hyperlink-Induced Topic Search (HITS) algorithm in particular, such as highest ranked search engine results predicated on the hyperlink configuration of the World Wide Web and analysing privacy in social networks in order to compute node weight and understand the elements of each object (endpoint) in the network. By using the HITS algorithm, we can promote POIs to LBSN users while simultaneously considering the influence of social ties. Our suggested model is tested on the Foursquare dataset and compared to the most recent POI recommendation algorithm. When we tested it against two prominent algorithms using real-world datasets, we discovered that our suggested approach performed better in terms of both variety and accuracy.

Keywords – Recommender Systems, Hypertext Induced Topic Search, Location Based Social Networks, Point of Interest.

I. INTRODUCTION

Smartphones and location-based social networks (LBSNs), also including Gowalla, Twitter, and Google Locations, have dramatically improved the lives of its users. These platforms allow users to check-in at POIs (Points of Interest) to show where and once they are, and to communicate their own perspectives with others through comments [1]. On Foursquare alone, more than 50 million users generated over 10 billion check-ins during the past year. Using so much data from check-ins, the question of how to extract user preferences and propose the correct POIs to the right consumers has become an important issue, which allows users to discover new destinations and facilitates network operators to launch adverts to potential customers. POI suggestion has received a lot of attention, with a variety of solutions being presented to solve the problem [2].

The challenge of consecutive POI suggestion, which suggests nearby POIs based on a user's current location and other contextual information, has become increasingly feasible and relevant as mobile devices make it easier to collect such information. After a person has eaten, it's more logical to suggest a recreation location than a gym [3]. Furthermore, if we can foresee the future POIs of users, we can figure out where the event will take place. This task, however, is more difficult than standard POI suggestion due to the following factors. There may be tens of thousands of possible next-check-in POIs for a single query (user, current location), even though interactions between users and POIs are extremely rare [4].

A user's personal choices and the current POI have a big role in determining what will be the next POI. After a day of hiking or other outdoor activities, it's simple to think that people would rather have supper than go shopping. As a result, the efficacy of consecutive POI recommendations depends on how to deal with sparse and sequential information [5]. Prospective itinerary identification, friend suggestion, direct marketing, and POI recommendation are only some of the uses of LBSN, which may be found in a wide range of situations [6]. It leverages past check-in data to model the user's behaviour and mine the user's preference for destinations. Using POI recommendations to improve the user experience and help marketers target consumers is a win-win for both parties [7]. **Fig 1** shows Authority and Hub Structure of A HITS System.

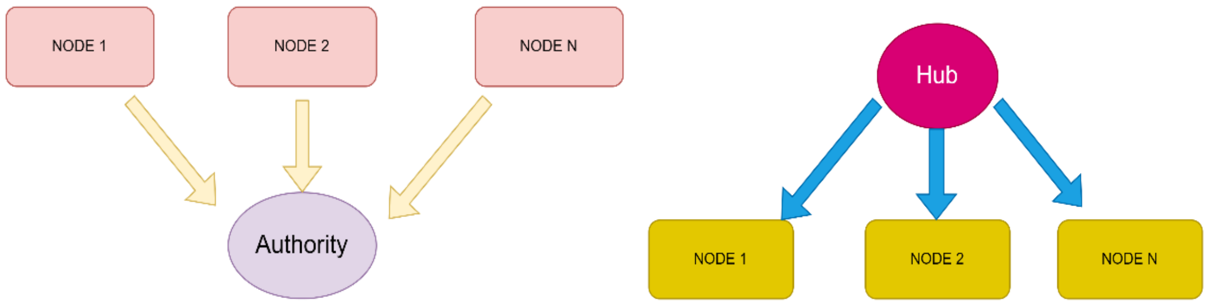

Fig 1. Authority And Hub Structure of A HITS System.

Even though users may share their location information at any time and from any place, the sheer volume of data makes it difficult for them to zero in on the ideal location for their needs. The POI recommendation service is designed to provide mobile users with customised suggestions of nearby locations [8]. Locations identified by their coordinates as well as utility tags including restaurants, movie theatres, and attractions are called POIs. Based on the check-in habits of comparable users, most POI recommendation systems learn about users' interests. Collective filtering (CF) is used in these algorithms since it believes the same people who like one service also prefer another. As a result, users can get suggestions for nearby points of interest based on the check-in data of their peers. It's important to keep in mind that people's behaviours might shift with the passage of time and place, and this approach fails to account for that [9].

Because most people visit different locations at various times of the day, this strategy may result in incorrect suggestions. For example. It's possible to identify POIs and share your own interactions and personal activities relating to a certain area through the usage of LBSNs. The most popular LBSNs, like Foursquare and Facebook Places, have millions of users. Since LBSNs have amassed such a wealth of personal preference and location data, many service providers are eager to make use of it for commercial gain [10]. For example, a retailer may use LBSNs to promote specials and discounts in order to draw in more consumers. LBSN users might also benefit from such extensive datasets, such as promoting POIs based on check-in behaviour. LBSN recommends content based on a variety of parameters, including location, time, social media, category, comments, and images [11].

To enhance the effectiveness of POI suggestion, such influential factors should be included and mined. A matrix factorization-based recommendation approach has been presented to fill in the user's choice for previously unvisited sites due to the lack of check-in behaviour. Check-in data reveals the preferences of distinct users over time and place for various geographical categories [12]. Filling in incomplete data of tensor user-category-time tensor is done by employing the tensor decomposition technique first. The sort of user's demand is shown by the location's categorization. The analysis of the user's preference category acquired from the tensor can be used to omit any unnecessary places. Users' preferred location is determined using the tensor decomposition result [13].

A user's choice for a place is then considered from two perspectives, including the computation of similarity and spatial constraint [14]. A new technique of location preference calculation based on user and temporal similarity augments the user-based collaborative filtering method. It is possible to reduce the number of computations by using only the top k individuals who are most appropriate for the target user. The other considers a user's ranking of the most popular spots during a specified time period. The location of the user's previous check-in locations is a consideration here. As a last step, the HITS algorithm is used to determine a location's popularity [15].

Because LBSNs users would visit POIs for the aforementioned variety of reasons, precisely and thoroughly capturing these motives is difficult. The first step is to evaluate our dataset to identify the attributes of venues that are likely to draw in more visitors [16]. There are more check-ins at popular POIs. However, a POI's popularity is not just determined by the number of people who have checked in, but also by the manner in which those people have checked in. The key contributions of this paper are as follows:

- We present a POI recommendation method based on the HITS model. Users' actions can be anticipated to impact their friends' activities in a similar way to how HITS-based models are improved.
- The suggested method is adaptable since the parameters may be changed by the user to tailor the advice. In addition, the algorithm can handle a wide range of weights.
- We can also use our technique to propose POIs to a group of users, which is beneficial for helping a group of friends/colleagues select a location for a get-together, for example.
- Our suggested POI recommendation method is tested using the Foursquare dataset. Using past research as a benchmark, we find that using entropy-weighted weights is the most effective strategy we could come up with.

II. LITERATURE SURVEY

The incorporation of spatial and temporal aspects into systems has been used in recent studies to increase the accuracy of POI recommendation systems. As an example, Yuan et al. suggested a time-aware POI recommender system to increase the accuracy of POI suggestions, and this system provides a list of POIs comparable to the vast majority of POI categories that the user frequently visits during specific time periods. The problem is that it tends to result in a homogenous group of POIs. To prevent POI recommendation systems from being homogenised, Chen et al. presented an information coverage" strategy, which takes into consideration consumers' preferences and the variety of service categories [17].

As a result of this strategy, consumers may be advised with additional POIs from a variety of categories [18]. Even if the POIs are related in some way, consumers may have difficulty selecting a few of them from a long list. Typically, users only pick one POI to access, that might result in a high number of incorrect suggestions. Despite the fact that consumers may have a wide range of interests, there is an inherent regularity in the time period in which they choose a service [19]. As an example, consumers may find it difficult to choose from a list of recommended POIs that includes tourist sites and retail malls as well as a zoo and Internet cafés, even if they may be involved throughout items.

POI collections that include "adventures, stores, and restaurants" are more popular with users than those that don't. "Souvenir stores" and "signature eateries" can be found in the vicinity of "attractions." As a result, it is important to analyse the links between POIs that are suggested in order to limit the number of incorrect suggestions [20]. SocialMF, developed by Jamali and Ester, incorporates the transmission of trust to enhance the precision of the recommendation. An algorithm developed by Cheng et al. uses probabilistic matrix segmentation and a social normalisation factor to make recommendations on where to go. It aims to enhance the effectiveness of the location suggestion algorithm by integrating more geographical impacts [21].

However, these approaches only suggest areas in which the user has shown an interest, regardless of how timely the information is. Recommendation systems may obtain less than ideal results if they know that a user like shopping in the afternoon, thus they should avoid recommending a certain store to that user in the morning [22]. When producing a suggestion for a user, it is important to consider both the preferences of the user and the time frame. When it comes to recommending POIs to a certain user at a specific time of day, the issue is how to do it. Consumer information retrieval POI recommendation approach by mining the effect of time frames and geographical features was presented by Yuan and colleagues [23].

Measurement of place popularity is defined by using a distance function. Tensor deconstruction is also used for time-aware POI suggestion, considering the temporal effect. Tensor factorization was used by Zhao and his colleagues to analyse the relationships between POIs, users, and POIs, and POIs and time [24]. Using a ranking technique for POI recommendation, Li and his colleagues suggested a fourth-order factorization based on tensor factorization to determine the effect of temporal change on user decision-making It also considers the long-term as well as the short-term preferences of the users [25].

Existing research attempts cannot increase performance since just one or two elements are used, whereas additional relevant information is needed more to improve the effectiveness of recommendations. With Berjani and Strufe, you can get individualised recommendations for POIs in LBSSNs using a regularised matrix factorization-based recommender [26]. To overcome this issue, they offer a user preference method based on check-in counts, which they say is the major obstacle of POI recommendation in the LBSN. They tested their suggested strategy using the Gowalla dataset, and the findings show that CF-based strategies may be used to provide POI suggestions. This paper examines the traditional trajectories as well as unusual locales, based on GPS data from travellers. Users and POIs may be linked using this data, allowing for better POI suggestions for passengers. For starters, they recommend utilising the HITS framework to describe relevant locations [27].

As a result, their technique does not consider the social impacts among users, as they employ GPS motions of specific users and it is difficult to trace the social relationships among these users [28]. Based on their findings, Zheng et al. suggest using GPS data to promote POIs and activities in collaboration. In their suggested technique, they exhibit improvements in POI and activity recommendations over the basic baseline by utilising the POI attributes and activity-activity correlations. Despite this, their methodology does not consider the social factors [29].

Leung et al. present a GPS-based framework for collaborative location recommendation (CLR). A user's location entropy is used to categorise them into three groups: Pattern users, Normal users, and Travelers. It uses a clustering technique known as CADC (Community-based Agglomerative-Divisive Clustering). The CLR is able to produce more accurate and refined suggestions based on the clusters. Urban POI-Mine (UPOIMine) is a technique proposed by Ying et al. to recommend POIs that considers both the interests of users and the qualities of the surrounding area [30]. Personal preferences, category context, highlight context, and POI popularity are all considered while producing suggestions.

LBSN socio-spatial aspects are examined by Scellato et al., as well as the role of location variables in link prediction in LBSNs. In order to anticipate future check-ins, Noulas et al. have extracted and studied the LBSNs users' movement attributes [31], [32]. Several elements are proposed to capture the aspects that may encourage users to check in again in the future. Then, in LBSNs, they suggest fresh POI recommendations based on a random walk. In the research, they look at how often LBSNs users visit new POIs and examine the assumptions made when employing web-filtering algorithms to forecast human mobility [33]. Researchers have found that current filtering algorithms do not yield high-quality suggestions, and instead offer tailored random walk recommendations based on the examination of the LBSNs dataset. On the Gowalla dataset, their trials reveal a 5-18% improvement with the proposed random-walk proposal [34].

III. PROPOSED SYSTEM

Before going into the specifics of our strategy, we'll go through the real-world dataset we have used, as well as some key trends in user behaviour we'll be considering for our model. During April 12, 2012 through February 16, 2013, Foursquare collected user check-in data in New York City. We removed points of interest (POIs) that had been visited by little more than five people and filtered individuals who had checked in at no more than ten POIs. A total of 17,816 points of interest (POI) are returned from Foursquare's database in and around Pittsburgh. These POIs fall into 9 major categories and 271 secondary categories. From March to July 2012, we tallied the check-ins at these POIs. Throughout this period, 44,437 Foursquare users made 1,226,769 check-ins at some of these POIs. Because of this, the U_{ac} is 27.61 check-ins per user, but the P_{ac} is 68.86. Among the 44,437 users, we also find 297,580 friendship relationships. For categorising POIs, Foursquare uses a hierarchical category system defined by the company.

System Model

Hubs and authorities, or HITS, is a term used to describe the process of extracting information from connection architectures. By assigning a page a hub and authority score, HITS determine which pages are the best sources of information on a topic. Because the hub values of pages linking to it are added together, a page's authority score is equal to the total of the hub values of the pages linking to it, and vice versa. A good authority page is linked to by a lot of excellent hubs and a good hub page points to a lot of good authority sites, which shows that hubs and authorities work hand in hand. In this, the basic update operations are of the two vectors defining hub(sv_h) and authority(sv_a) scores are done as:

$$\begin{cases} sv_h = M_a . sv_a \\ sv_a = {M_a}^T . sv_h \end{cases} \tag{1}$$

Our suggested POI recommendations in LBSN is based on the HITS algorithm's ideas of hubs and authority. It is our goal for LBSN users to see POIs with better authority scores; this is based on their hub scores. HITS, on the other hand, cannot be directly used to POI suggestions in LBSNs since the original HITS algorithm tries to tackle online search problems and social impact concerns are not relevant in web search problems. In LBSNs, users' POI visits are likely to be influenced by their social connections as well. There are a number of ways that a user's friends might propose POIs to him that he may like to visit, such as by checking in at the same POI together or by recommending some other POIs to the user. As a result, POI suggestions should also consider social relationships. We refer to the ratings generated by the check-in statistics of users at different times and locations as the relevance assessment in the recommendations. **Fig 2** shows Proposed Model Architecture.

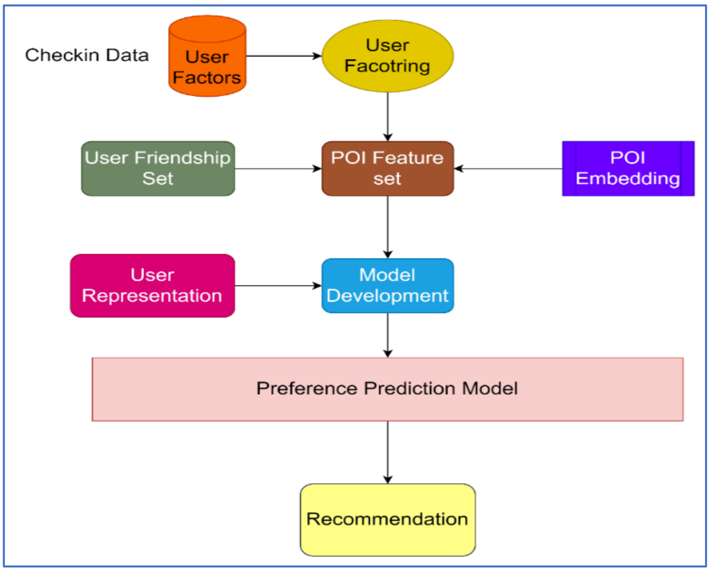

Fig 2. Proposed Model Architecture.

Users may be interested in POIs in the relevance-based recommendation set, but the POI set may lack associations between POIs, making it impossible for users to access the POIs over and over again. This component examined the relationships among POIs to find the most effective pathways and developed a list of POIs covering more effective paths from which users may pick, thereby encouraging the user to visit more POIs. To study the relationships between POIs, it is necessary to identify the POI association rules. A POI's association rules may be mined to determine how different POIs are connected because they are all designated with the same service tag. There are service tags for each POI, as well as the location property. This means that service types and locations influence the relationships between POIs. Considering that the location of POIs has a comparable effect on the associations among POIs, the relevance recommendation algorithm takes the location element into account and typically places POIs within a 10 km range. As a result, we'll use this section to determine the relationships between POIs depending on different services they offer.

Notations and System Considerations

In the proposed network structure, we are assuming a graph structure for representing the LBSN structure. It is important to first explain the network paradigm that underlies our suggested HITS-based POI recommendation framework. Check-ins at POIs that users have built and friendships they've formed are both represented as nodes in the network. We define the graph structure as $SN_g = (SN_v, SN_e)$ and the components of the graph are defined as follows:

$$SN_v = \{SN_{v1}, SN_{v2}, \dots SN_n, SN_{n+1}, SN_{n+2}, \dots SN_{n+m}\} \tag{2}$$

$$SN_e(x, y) = SN_v(x) \leftarrow SN_v(y) \tag{3}$$

where SN_v denotes the vertices comprising of the users and their check-ins and SN_e represents the edge in the social network graph that is represented by SN_g.

$$\forall x, y\ SN_v(x) \leftarrow SN_v(y) \in SN_e \text{ and } SN_v(x) \in SN_v \text{ for } 1 \leq x\ and\ y \leq n + m \tag{4}$$

In this the value of n is obtained from the overall count of the users of the considered network and m represents the overall count of the POIs in the network. They could be represented as:

$$SN_{user} = \{SN_{v1}, SN_{v2}, \dots SN_n\}\ and\ SN_{poi} = \{SN_{n+1}, SN_{n+2}, \dots SN_{n+m}\} \tag{5}$$

The entire set of the considered network contains $|SN_v|$ number of vertices in combination, and can be obtained from the sum of $n\ and\ m$. In the social network graph, we consider different friendship edges that are represented by an undirected edge among the users and the POI. From this, we can obtain the adjacency matrix as follows:

$$M_a^{n\times n} = \begin{cases} 1, while\ SN_e(x, y) \in SN_e \\ 0, If\ else. \end{cases}, such\ that\ 1 \leq x, y \leq n. \tag{6}$$

Also, we are considering check-in nodes that are in the graph and interested in calculating the corresponding adjacency matrix as:

$$M_h^{n\times m} = \begin{cases} 1, while\ SN_e(x, y) \in SN_e \\ 0, If\ else. \end{cases}, such\ that\ 1 \leq x \leq n\ and\ n + 1 \leq y \leq n. \tag{7}$$

We could define our proposed model for recommendation system based on POI and HITS with the help of the friendship and check-in edges as:

$$POI_{sv_a} = (1 - \alpha) POI_w . U_{sv_h} \tag{8}$$

$$U_{sv_h} = (1 - \alpha) U_w + (1 - \alpha) POI_w . U_{sv_a} + \alpha U_{sv}^0 \tag{9}$$

There are three factors that define a POI's authority score: users who already have checked in at the POI, the hub ratings of their friends, and the authority scores of the POIs that the user has logged in at. Users will be given recommendations for POIs based on the recommendation computation authority scores. The algorithm's weights will be defined as follows:

We begin by looking at the graph's simplest case, which has a uniform distribution of edge weights. Overall weights for the friendship edges inside the social graph are defined as follows:

$$U_w = \begin{cases} \frac{1-\beta}{d(SN_{vx})}, while\ d(SN_{vx}) > 0\ and\ Count_{vx} > 0 \\ \frac{1}{d(SN_{vx})}, while\ d(SN_{vx}) > 0\ and\ Count_{vx} = 0 \end{cases} \tag{10}$$

where $d(SN_{vx})$ denotes the user degree for the social network with the friendship edges for the different POI values.

Overall trustworthiness of a friendship edge may be used to determine the weight of an edge inside the social graph. If such confidence values among users were available, we would apply the same friendship edge weights as in our dataset. We present two techniques for estimating the weights of edges among POIs and users. The check-in counts are used in the first technique to establish weights, whereas the entropies are used in the second way.

Proposed Recommendation Process

To determine the relative importance of online sites, the HITS algorithm does link analysis. Authorities and hubs are two of the most important notions in the algorithm. Having a large number of inbound connections indicates that a web page is trustworthy and contains useful information, which is indexed by a large number of other web pages. A hub section is a web page that acts as a clearinghouse for information on a particular topic and provides links to other authoritative resources. An authority is assumed to be cited by a large number of hubs, while a hub is assumed to be mentioned by numerous authorities.

As with user trustworthiness and service reputation, a user can rate numerous services, and a service can get feedback ratings from many users. The subgraphs directed Graph was introduced by applying this concept to it. The aggregate of all the ratios between the user feedback rating and the reputation of each service, as determined in the previous incarnation, is the credibility of the user. The user's credibility rises if the feedback is near to what the consensus has ascribed to the online service, which is reflected by the reputation. As previously stated, the HITS approach is not adequate for analysing today's complex graphs since it treats each edge as if it were a separate entity.

As a result, the evaluation of edges and the computation of weights are central to our work on improving HITS. Typically, the value of an edge is determined by the coincidences or correlations in the content or internal qualities of the two endpoints that make up the edge itself. Since the specifics of each graph must be considered, this approach cannot be used for all types of graphs. Our goal is to develop a worldwide approach for edge evaluation, and our strategy is to consider the number of an edge simply based on the hub and authority of two end points. An important aspect of this study is to determine which of these two factors has a direct impact on a particular group-based edge's authority and relevance. This association serves as the basis for evaluating the group-based attribute.

Algorithm 1: Recommendation algorithm based on POI user and Time

Input: $\boldsymbol{U}, \boldsymbol{T}, \boldsymbol{M_{chkin}}$

Output: $\boldsymbol{Rec_{U,T}}$

1. Estimate $\boldsymbol{SN_v = \{SN_{v1}, SN_{v2}, \dots SN_n, SN_{n+1}, SN_{n+2}, \dots SN_{n+m}\}}$.

2. $\boldsymbol{\forall x, y}$ **do**

Find $\boldsymbol{M_a^{n\times n} = \begin{cases} 1, while\ SN_e(x,y) \in SN_e \\ 0, If\ else. \end{cases}}$.

3. Estimate $\boldsymbol{POI_{sv_a} = (1-\alpha) POI_w . U_{sv_h}}$

4. Find $\boldsymbol{U_{sv_h} = (1-\alpha) U_w + (1-\alpha) POI_w . U_{sv_a} + \alpha U_{sv}^0}$.

5. From the estimated $\boldsymbol{\langle POI_{sv_a}, U_{sv_h} \rangle}$

$$\boldsymbol{U_w = \begin{cases} \frac{1-\beta}{d(SN_{vx})}, while\ d(SN_{vx}) > 0\ and\ Count_{vx} > 0 \\ \frac{1}{d(SN_{vx})}, while\ d(SN_{vx}) > 0\ and\ Count_{vx} = 0 \end{cases}} \mathbf{x}$$

6. Compute $\boldsymbol{Rec_{U,T}}$ **as** $\boldsymbol{\sum_{x,y \le n+m} and\ M_h^{n\times m} = \begin{cases} 1, while\ SN_e(x,y) \in SN_e \\ 0, If\ else. \end{cases}}$.

HITS describe hub as a measurement of the value of its interconnections to other nodes, and this cost is distributed evenly to its outgoing edges. To put it another way, authority may be used as a way to estimate the worthiness of the edges that connect to it, and the value is distributed evenly among them. Nodes in the same subject or group should be evaluated equally in analysis, and each graph edge should be estimated at the same level of significance. Furthermore, when all edges in a graph are homogenous, their two endpoints are in the identical group and their value is similar, hubs and authorities have a perfect correlation when their allocation values are unified into a single edge. Connections between nodes that belong to various groups are referred to as a none group-based edge.

A considerable discrepancy between these assigned values is expected for none group-based edges, which connect nodes inside a single group. This discrepancy, we assume, is much greater than for group-based edges. The group-based feature of edge is evaluated using the in correlative rate as a measurement and foundation. Correlation and “group-based” characteristics are more important for edges that have lower correlation rates, and vice versa. For this, we conduct experiments and examine the correlation rates rij of regular and group-based edges. For the sake of experimentation, we employ link-farming as a representative sample of the non-group-based edge. Out-going and in-coming link-farms are two types of link-farms.

As a result, a link-destination farms may have a high in-degree, but minimal connection to the graph's remainder. It's possible that spam nodes and link farms have a high out-degree, but they're not connected to the rest of the graph in any way. Our investigation compares the in correlative rates of edges via sample spamming on graphs. Due to the fact that edges on a real network cannot be judged for their honesty, we utilise a random graph in which all edges are generated at random, simulating association in a random graph through a random function. On the other hand, as we know, the value of edges is assigned based on the value of two ends. In an effort to standardise values, we recommend using the weight of pertinent edges with the same source or destination. Distinct normalisation operations are often carried out for edges that have a common source and a common destination.

Algorithm 2: HITS based Recommendation with Social network graph

Input: $SN_g = (SN_v, SN_e)$

Output: Recommendation Nodes.

1. **Set initial weight values of** SN_h, SN_a *and* SN_w **as specified in the problem.**
2. **Estimate** $\overline{SN_w}$ **and the value of** $\rho(\overline{SN_w})$.
3. $\forall\ SN_{node}$ **do,**
   a. **Estimate** $SN_h = \sum_{x=1}^{l} SN_{wx} . SN_a$
   b. **Also, find** $SN_a = \sum_{y=1}^{x} SN_{xy} . SN_h$
4. **Normalise the hub and the authority information as** $SN_w(x,y) - \Delta(SN_w(x,y) \leq 0$.
5. **Calculate the corresponding edge weight as,**
   $SN_w(x,y) = SN_w(x,y) - |SN_w(x,y).(SN_h - SN_a)|$, **while** $x \neq y$ *and* $SN_w(x,y) \neq 0$.
   $SN_w(x,x) = SN_w(x,y) + |SN_w(x,y).(SN_h - SN_a)|$, **while** $SN_w(x,x) \leq \overline{SN_w} + \rho(\overline{SN_w})$
6. **Normalise the edge weights as**
   $$SN_w(x,y) = \frac{SN_w(x,y)}{\sqrt{\sqrt{\sum_{x=1}^{l}(SN_{wx})^2} . \sqrt{\sum_{y=1}^{x}(SN_{xy})^2}}}$$
7. **Return recommended Authority hub list.**

For each user's present location, our algorithm generates a prioritised list of the next few POIs they would like to visit. However, the ranking sequence of candidate POIs is more important to us than the probabilities. We present a pairwise ranking objective function in accordance with the BPR optimization criterion. According to user preferences, we provide recommendations for nearby locations. Locations that don't fit the user's interests should be removed from the user's POI suggestion list. Based on this, the check-in frequency matrix is rebuilt. As a further step, we gather the user's preferred places at a given time. However, unlike existing collaborative filtering methods, we build a similarity computation algorithm that considers both the time and the user.

Final findings are returned as recommendations after the user-based collaborative filtering process has been used to obtain probability values for all POIs. We employ a collaborative filtering technique and a new similarity computing approach to calculate the user's preference for each place. Rather of calculating the degree to which the target user as well as all other users are alike, this study examines just those users who have a high degree of similarity with the target user in terms of their actions. In this case, we employ a collaborative filtering technique to pick out people with high similarity values and exclude those with poor similarity values. Using check-in information, we can partition every day across time frames based on the time spent at each location.

By combining the cosine similarity metric with different time windows, we are able to compare the location-visiting patterns of all users. For sparse data and better recommendation time correlation, we mix the likelihood of time interval with recommendation methods and apply flattening technology extended to all time slots. Calculating how likely it is for a user to visit an area at the given time slot requires us to determine the degree to which the user has previously visited the place and then multiply that number by the proportion of likeness between that location and the overall similarity. In the end, a POI recommendation result may be generated based on E, which signals a new choice value both regarding the time slot and user's similarity. The preferred rating of geography for a target user is calculated using just the upper similarity between users to that target user. Users with a high degree of similarity play a vital part in suggestion.

The distribution of users' check-in locations is initially examined for POI suggestion using graphical data. Users tend to congregate in a specific region when they visit the site. As a user, the check-in sites constitute a cluster for you. This is an indication that the user is focused on a certain region. A user can go to a new location that is near to where he has previously checked in. To put it another way, the user is much more likely to be interested in a location that is close by. For example, if a user is looking for a nearby place, we can propose it based on the user's location. We get a pseudo centre of a user's past visiting places and calculate the distance between each site and the pseudo centre for each user.

Distance between two points can be calculated using the great circle technique, which measures the length of a straight line that connects them. When it comes to location, people are much more concerned in what is close by. They like to visit a place that is near to home. Users' most frequented destinations may also include useful preference data for your intended audience. As a result, places that are both well-known and conveniently located for a user's current location will be given consideration as potential destinations. In order to determine the popularity of the sites, this study makes use of HITS. If a place is well-liked by the general public, it will be treated as an authority page, with each user as its own hub page. For given user at time slot T, we receive a list of possible locations based on geographical limitations and the HITS algorithm.

IV. RESULTS AND DISCUSSION

Based on our Foursquare dataset, we test out the HITS method described in this section. Before presenting our findings, we explain the methods and metrics we used to assess the algorithms. First, we remove users and POIs with a small number of check-in events from the dataset, and then we randomly partition the remaining dataset into two sets: one for testing algorithms and the other for training them. In particular, we utilise the authority ratings to rank the POI options after running the suggested algorithms on the network produced from the training dataset. Depending on the rankings of the POIs, we propose the top-N POIs.

We next compare the top-N POIs to the testing dataset. There is a unique identifier for each user and a unique POI-ID for each place. Users that checked in less than five POIs and POIs that were verified by fewer than five users were omitted from the recommendation algorithm due to the lack of reference value. Following standardization, the Foursquare dataset includes 289306 check-ins supplied by 1321 people, and it contains 4412 POIs. To test the effectiveness of the proposed method, we carried out an experiment to evaluate the propagation path reportage top-POI recommending system with multiple state-of-the-art recommendation algorithms. Both the UST and the greedy algorithm (GA) with pruning optimization are used to determine the top-k LC-POIs, or user-spatial-temporal unified frameworks. The following three metrics were used to evaluate the quality of the recommendation approaches.

The studies were carried out on a Windows PC with a 4-core Intel i5 3.2 GHz CPU and 8 GB RAM using Java and the experimental environment. For our suggested technique, the specific aspects employed for model are given as follows. Node benefits for the user include number of check-ins, unique visited POI count and check-in timing information like day-of-week, time of day and social connections. Features of POI nodes, include information on POIs' categories and latitude and longitude, which are handled via one-hot encoding. User nodes and POIs may also be connected by edge features like how many times they've visited each other's POIs, how many times they've checked in to each other's POIs on the same days/hours, and how many times they've checked in to each other. **Table 1** shows Precision Comparison with Existing Algorithms.

Table 1. Precision Comparison with Existing Algorithms

		Precision	
N	UST	AKAWO	Proposed HITS
1	0.08	0.1254	0.1459
2	0.0768	0.1211	0.1412
3	0.0725	0.1196	0.1392
4	0.0701	0.1168	0.1356
5	0.0687	0.1124	0.1322
6	0.0654	0.1089	0.01254
7	0.0639	0.1035	0.1212
8	0.0611	0.0965	0.1985
9	0.0602	0.0865	0.1886
10	0.0589	0.0785	0.1725

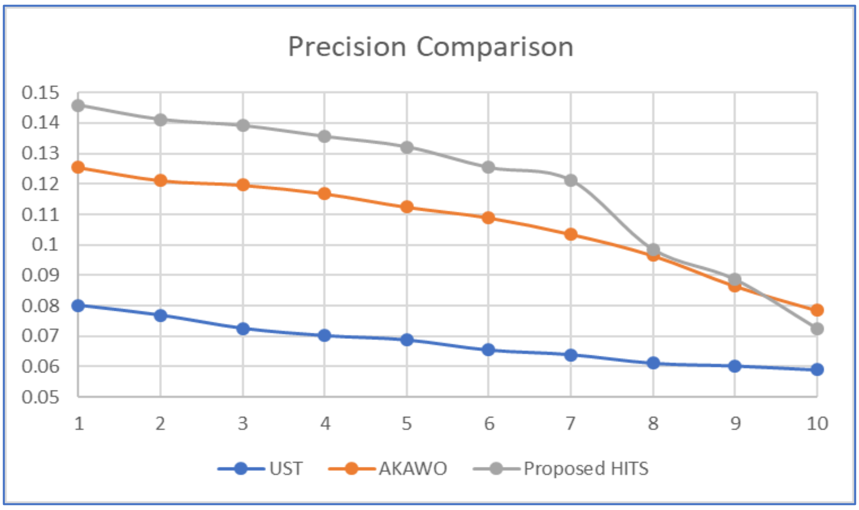

Fig 3. Precision Comparative Analysis.

In order to evaluate a recommendation model's effectiveness, we use three metrics: precision, recall and normalised discounted cumulative gain which assigns larger points to hits at the top positions. If you don't specify differently, the top-

10 recommendation is applied for calculating metrics. The findings of each experiment are averaged over a total of ten trials. In this part, we test the model of the proposed approach from four perspectives, namely, negative sampling, social ties, sequential check-in behaviour, and profile information. **Fig 3** shows Precision Comparative Analysis.

The framework's hyperparameters are set as follows: User and POI embedding dimensions are both set to 128. The hidden state feature dimensions are both set to 32. POI sequence length is by default eight bytes. The user node representation vector sizes are both set to 32, and MLP depth, which determines preference score, is set to two by default. How social media network information effects the operation of the framework and the attention mechanism influences the combined with social links are two ways in which social interactions are evaluated. To demonstrate the effectiveness of modelling social impact we observe that if a basic average technique is applied, the framework enhances the variant without using social network information. When it comes to combining social influence, we also look at the structure of the attention strategy in use. **Table 2** shows Average Precision Value for Different Parameter Combinations.

Table 2. Average Precision Value for Different Parameter Combinations

N	β = 0.05 and λ = 0.15	β = 0.25 and λ = 0.75	β = 0.35 and λ = 0.65	β = 0.5 and λ = 0.75	β = 0.85 and λ = 0.95
1	0.215	0.205	0.195	0.185	0.158
2	0.209	0.199	0.189	0.179	0.112
3	0.2014	0.1914	0.1814	0.1714	0.0896
4	0.1992	0.1892	0.1792	0.1692	0.0758
5	0.1895	0.1795	0.1695	0.1595	0.0724
6	0.1756	0.1656	0.1556	0.1456	0.0711
7	0.1625	0.1525	0.1425	0.1325	0.0702
8	0.1598	0.1498	0.1398	0.1298	0.0668
9	0.1568	0.1468	0.1368	0.1268	0.0621
10	0.1521	0.1421	0.1321	0.1221	0.06

We are particularly interested in how the attention mechanism's performance is affected by the network's depth. Because sequential check-in behaviour patterns are so important to users, we found that the entire framework considerably outperformed the variation in all situations. Social network information has a less impact on tailored POI suggestion, suggesting that sequential activity patterns have a more significant influence. Because we explicitly model users' sequential check-in habits, we take into consideration both spatial and temporal influences, as well as the effect of these factors on the model's output. **Fig 4** shows Parameter Based Precision Analysis.

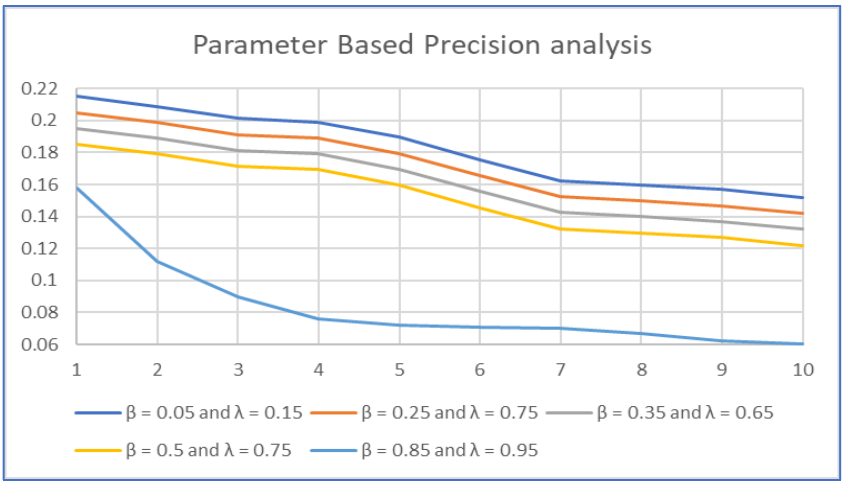

Fig 4. Parameter Based Precision Analysis.

The complete dataset is used in this part. We'll use β = 0.85 and λ = 0.95 as our starting points, and then we'll look at the parameters. Compared to previous algorithms, the suggested approach employing entropy weights has the highest average precision, followed by the HITS-based algorithm utilizing the uniform weights. Recall is greatest for entropy weights in the proposed method, followed by uniform weights in HITS in the second place. All three new algorithms outperform older ones in terms of precision and recall. The recall performance of all four methods improves with larger values of the constant N. Using the same settings, we will test the accuracy and recall in the Evening Spot category dataset in this subsection. **Table 3** shows Recall Value Comparison.

Table 3. Recall Value Comparison

	Recall		
N	**UST**	**AKAWO**	**Proposed HITS**
1	0.069	0.05	0.1
2	0.079	0.1	0.15
3	0.115	0.129	0.2
4	0.136	0.165	0.23
5	0.168	0.198	0.27
6	0.245	0.254	0.3
7	0.285	0.267	0.353
8	0.312	0.298	0.39
9	0.322	0.352	0.321
10	0.335	0.379	0.365

Entropy weights have the highest average accuracy, followed by respect to the weights based on check-ins, according to our results compared to other suggested algorithms. It's also clear from this graph that the suggested algorithm based on entropy weights, backed by the recommended algorithms based on the check-ins, is the most accurate. Although there are no significant differences between the suggested algorithms employing different edge weights, recall rises as N grows. Both of them outperform the suggested method, which uses uniform weights, by a large margin. It's important to note that, when applied to the complete dataset, the uniform weights technique outperforms the check-in weights algorithm. **Fig 5** shows Recall Comparison Analysis.

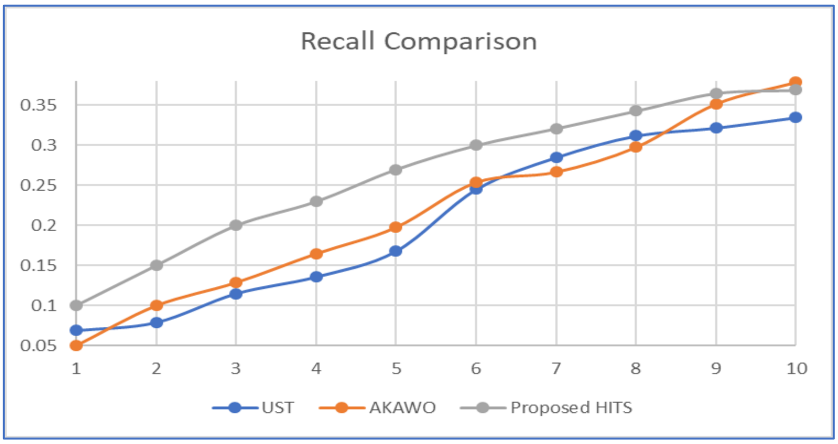

Fig 5. Recall Comparison Analysis.

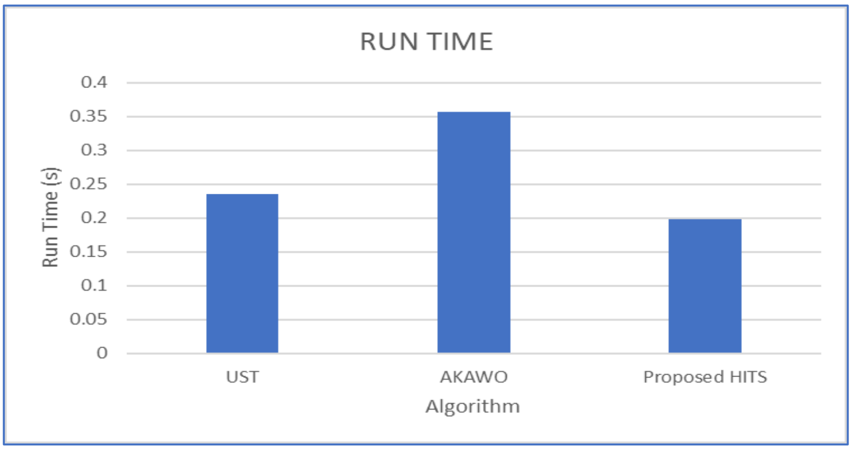

Fig 6. Runtime Comparative Analysis.

Users' check-ins are more diversified when compared to a single POI's classified data, and the category information adds additional uncertainty to the recommendations. Because of this, the weights should be evenly distributed. In this section, we examine the effects on our dataset of varying the values of the parameters and. A dataset is used to demonstrate the accuracy of the proposed method with entropy-based weights. $\beta = 0.85$ and $\lambda = 0.95$ have the highest precision, whereas $\beta = 0.15$ and $\lambda = 0.95$ have the poorest precision. The original hub score vector's weight is determined by the value of the parameter. **Fig 6** shows Runtime Comparative Analysis.

Because the original hub score vector for a user is zero except for the k^{th} element, which has a 1 in it, the nodes that are closest to the user are more important when updating hub scores, we found this to be true in our experiment. It is more difficult to configure parameter. When updating hub scores, social influence and check-in behaviour are balanced out by the parameter. The higher the, the greater the impact of check-ins on hub score updates. The greater the value of, the poorer the outcomes; however, when is significant (example, β= 0.85), the greater the value of is, the better results are obtained. In summary, the best values for and should be determined depending on the dataset's characteristics.

V. CONCLUSION

In LBSN, POI suggestion is a popular study focus. LBSNs allow users to share and locate POIs, as well as their own social activities and memories associated with specific areas. With the use of social network ties and check-in behaviour from LBSN users, we developed a HITS-based POI recommendation algorithm for usage in LBSNs. Using two metrics – accuracy and recall – we compare our proposed algorithms with the most recent POI suggestion on the Foursquare dataset. The suggested approach with weights based on entropy provides improved results for both precision and recall. We also test the suggested algorithms on a classified dataset, and we find that the algorithms perform better on this dataset than the complete dataset does. The category information helps the recommendation work better therefore this is what we conclude. For further information on how different factors affect performance, please see the following section: In the future, we want to do quantitative research on the topic of edge weights and how category information might be utilised to enhance algorithms.

CRediT Author Statement

The authors confirm contribution to the paper as follows:

Conceptualization: John Vaseekaran S and Srinivasan N; **Methodology:** John Vaseekaran S; **Software:** John Vaseekaran S and Srinivasan N; **Data Curation:** John Vaseekaran S; **Visualization:** Srinivasan N; **Investigation:** John Vaseekaran S and Srinivasan N; **Supervision:** Srinivasan N; **Validation:** John Vaseekaran S and Srinivasan N; **Writing- Reviewing and Editing:** John Vaseekaran S and Srinivasan N; All authors reviewed the results and approved the final version of the manuscript.

Data Availability

No data was used to support this study.

Conflicts of Interests

The author(s) declare(s) that they have no conflicts of interest.

Funding

No funding agency is associated with this research.

Competing Interests

There are no competing interests

References

[1]. H. Ying et al., "Time-aware metric embedding with asymmetric projection for successive POI recommendation," World Wide Web, vol. 22, no. 5, pp. 2209–2224, Jun. 2018, doi: 10.1007/s11280-018-0596-8.

[2]. D. Yu, W. Wanyan, and D. Wang, "Leveraging contextual influence and user preferences for point-of-interest recommendation," Multimedia Tools and Applications, vol. 80, no. 1, pp. 1487–1501, Sep. 2020, doi: 10.1007/s11042-020-09746-0.

[3]. S. Wu, Y. Zhang, C. Gao, K. Bian, and B. Cui, "GARG: Anonymous Recommendation of Point-of-Interest in Mobile Networks by Graph Convolution Network," Data Science and Engineering, vol. 5, no. 4, pp. 433–447, Jul. 2020, doi: 10.1007/s41019-020-00135-z.

[4]. T. Bao, L. Xu, L. Zhu, L. Wang, and T. Li, "Successive Point-of-Interest Recommendation with Personalized Local Differential Privacy," IEEE Transactions on Vehicular Technology, vol. 70, no. 10, pp. 10477–10488, Oct. 2021, doi: 10.1109/tvt.2021.3108463.

[5]. Z. Cai, G. Yuan, S. Qiao, S. Qu, Y. Zhang, and R. Bing, "FG-CF: Friends-aware graph collaborative filtering for POI recommendation," Neurocomputing, vol. 488, pp. 107–119, Jun. 2022, doi: 10.1016/j.neucom.2022.02.070.

[6]. Md. A. Islam, M. M. Mohammad, S. S. Sarathi Das, and M. E. Ali, "A survey on deep learning based Point-of-Interest (POI) recommendations," Neurocomputing, vol. 472, pp. 306–325, Feb. 2022, doi: 10.1016/j.neucom.2021.05.114.

[7]. C. Zheng, D. Tao, J. Wang, L. Cui, W. Ruan, and S. Yu, "Memory Augmented Hierarchical Attention Network for Next Point-of-Interest Recommendation," IEEE Transactions on Computational Social Systems, vol. 8, no. 2, pp. 489–499, Apr. 2021, doi: 10.1109/tcss.2020.3036661.

[8]. G. Zhou, S. Zhang, Y. Fan, J. Li, W. Yao, and H. Liu, "Recommendations based on user effective point-of-interest path," International Journal of Machine Learning and Cybernetics, vol. 10, no. 10, pp. 2887–2899, Jan. 2019, doi: 10.1007/s13042-018-00910-5.

[9]. M. Yin, Y. Liu, X. Zhou, and G. Sun, "A tensor decomposition based collaborative filtering algorithm for time-aware POI recommendation in LBSN," Multimedia Tools and Applications, vol. 80, no. 30, pp. 36215–36235, Sep. 2021, doi: 10.1007/s11042-021-11407-9.

[10]. J. Zhang, X. Liu, X. Zhou, and X. Chu, "Leveraging graph neural networks for point-of-interest recommendations," Neurocomputing, vol. 462, pp. 1–13, Oct. 2021, doi: 10.1016/j.neucom.2021.07.063.

[11]. Tibermacine, C. Tibermacine, and M. L. Kerdoudi, "Reputation Evaluation with Malicious Feedback Prevention Using a HITS-Based Model," 2019 IEEE International Conference on Web Services (ICWS), pp. 180–187, Jul. 2019, doi: 10.1109/icws.2019.00039.

[12]. L. Feng, Y. Cai, E. Wei, and J. Li, "Graph neural networks with global noise filtering for session-based recommendation," Neurocomputing, vol. 472, pp. 113–123, Feb. 2022, doi: 10.1016/j.neucom.2021.11.068.
[13]. K. Baranitharan et al., "A collaborative and adaptive cyber défense strategic assessment for healthcare networks using edge computing," Healthcare Analytics, vol. 3, p. 100184, Nov. 2023, doi: 10.1016/j.health.2023.100184.
[14]. J. S. Kim, J. W. Kim, and Y. D. Chung, "Successive Point-of-Interest Recommendation with Local Differential Privacy," IEEE Access, vol. 9, pp. 66371–66386, 2021, doi: 10.1109/access.2021.3076809.
[15]. L. Chen, J. Cao, Y. Wang, W. Liang, and G. Zhu, "Multi-view Graph Attention Network for Travel Recommendation," Expert Systems with Applications, vol. 191, p. 116234, Apr. 2022, doi: 10.1016/j.eswa.2021.116234.
[16]. X. Sha, Z. Sun, and J. Zhang, "Hierarchical attentive knowledge graph embedding for personalized recommendation," Electronic Commerce Research and Applications, vol. 48, p. 101071, Jul. 2021, doi: 10.1016/j.elerap.2021.101071.
[17]. Z. Sun, C. Li, Y. Lei, L. Zhang, J. Zhang, and S. Liang, "Point-of-Interest Recommendation for Users-Businesses with Uncertain Check-ins," IEEE Transactions on Knowledge and Data Engineering, vol. 34, no. 12, pp. 5925–5938, Dec. 2022, doi: 10.1109/tkde.2021.3060818.
[18]. M. R and K. Komala Devi, "Food Classification by extracting the important features using VGGNet based Models in Precision Agriculture," 2024 2nd International Conference on Networking and Communications (ICNWC), pp. 1–7, Apr. 2024, doi: 10.1109/icnwc60771.2024.10537499.
[19]. G. Liao, X. Deng, C. Wan, and X. Liu, "Group event recommendation based on graph multi-head attention network combining explicit and implicit information," Information Processing & Management, vol. 59, no. 2, p. 102797, Mar. 2022, doi: 10.1016/j.ipm.2021.102797.
[20]. F. Zhou, T. Wang, T. Zhong, and G. Trajcevski, "Identifying user geolocation with Hierarchical Graph Neural Networks and explainable fusion," Information Fusion, vol. 81, pp. 1–13, May 2022, doi: 10.1016/j.inffus.2021.11.004.
[21]. S. Hosseini, H. Yin, X. Zhou, S. Sadiq, M. R. Kangavari, and N.-M. Cheung, "Leveraging multi-aspect time-related influence in location recommendation," World Wide Web, vol. 22, no. 3, pp. 1001–1028, May 2018, doi: 10.1007/s11280-018-0573-2.
[22]. R. Dridi, L. Tamine, and Y. Slimani, "Exploiting context-awareness and multi-criteria decision making to improve items recommendation using a tripartite graph-based model," Information Processing & Management, vol. 59, no. 2, p. 102861, Mar. 2022, doi: 10.1016/j.ipm.2021.102861.
[23]. K. Seyedhoseinzadeh, H. A. Rahmani, M. Afsharchi, and M. Aliannejadi, "Leveraging social influence based on users activity centers for point-of-interest recommendation," Information Processing & Management, vol. 59, no. 2, p. 102858, Mar. 2022, doi: 10.1016/j.ipm.2021.102858.
[24]. Y. Ying, L. Chen, and G. Chen, "A temporal-aware POI recommendation system using context-aware tensor decomposition and weighted HITS," Neurocomputing, vol. 242, pp. 195–205, Jun. 2017, doi: 10.1016/j.neucom.2017.02.067.
[25]. P. Balaji, K. Srinivasan, R. Mahaveerakannan, S. Maurya, and T. R. Kumar, "Swarm-based support vector machine optimization for protein sequence-encoded prediction," International Journal of Data Science and Analytics, Apr. 2024, doi: 10.1007/s41060-024-00551-8.
[26]. L. Chen, T. Xie, J. Li, and Z. Zheng, "Graph Enhanced Neural Interaction Model for recommendation," Knowledge-Based Systems, vol. 246, p. 108616, Jun. 2022, doi: 10.1016/j.knosys.2022.108616.
[27]. T. T. Hoa and N. N. Ha, "Edge-weighting Hyperlink-Induced Topic Search (E-HITS) Algorithm," Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017, pp. 925–930, Jul. 2017, doi: 10.1145/3110025.3110111.
[28]. X. Meng, Y. Tang, and X. Zhang, "DP-POIRS: A Diversified and Personalized Point-of-Interest Recommendation System," 2017 IEEE International Conference on Data Science and Advanced Analytics (DSAA), pp. 332–333, Oct. 2017, doi: 10.1109/dsaa.2017.24.
[29]. Z. Yu, H. Xu, Z. Yang, and B. Guo, "Personalized Travel Package With Multi-Point-of-Interest Recommendation Based on Crowdsourced User Footprints," IEEE Transactions on Human-Machine Systems, vol. 46, no. 1, pp. 151–158, Feb. 2016, doi: 10.1109/thms.2015.2446953.
[30]. E. Naserian, X. Wang, K. P. Dahal, J. M. Alcaraz-Calero, and H. Gao, "A Partition-Based Partial Personalized Model for Points-of-Interest Recommendations," IEEE Transactions on Computational Social Systems, vol. 8, no. 5, pp. 1223–1237, Oct. 2021, doi: 10.1109/tcss.2021.3064153.
[31]. P. Symeonidis, L. Kirjackaja, and M. Zanker, "Session-based news recommendations using SimRank on multi-modal graphs," Expert Systems with Applications, vol. 180, p. 115028, Oct. 2021, doi: 10.1016/j.eswa.2021.115028.
[32]. R. Gao et al., "Exploiting geo-social correlations to improve pairwise ranking for point-of-interest recommendation," China Communications, vol. 15, no. 7, pp. 180–201, Jul. 2018, doi: 10.1109/cc.2018.8424613.
[33]. D. Yu, T. Yu, Y. Wu, and C. Liu, "Personalized recommendation of collective points-of-interest with preference and context awareness," Pattern Recognition Letters, vol. 153, pp. 16–23, Jan. 2022, doi: 10.1016/j.patrec.2021.11.018.
[34]. J. Wang, H. Xie, F. L. Wang, L.-K. Lee, and O. T. S. Au, "Top-N personalized recommendation with graph neural networks in MOOCs," Computers and Education: Artificial Intelligence, vol. 2, p. 100010, 2021, doi: 10.1016/j.caeai.2021.100010.