INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

RESEARCH ARTICLE

A Repartitioning Temporary Customized Model for Interest-Point Recommendations with Hyperlink-Induced Topic Search

John Vaseekaran[1]*, **N Srinivasan**[2]

1 Research Scholar, Sathyabama Institute of Science and Technology, Chennai, India
2 Professor, Department of Computer Science and Engineering, Rajalakshmi Engineering College, Chennai, India

OPEN ACCESS

Received: 09-10-2022

Accepted: 30-11-2022

Published: 04-03-2023

Citation: Vaseekaran J, Srinivasan N (2023) A Repartitioning Temporary Customized Model for Interest-Point Recommendations with Hyperlink-Induced Topic Search. Indian Journal of Science and Technology 16(5): 339-347. https://doi.org/10.17485/IJST/v16i5.1996

* **Corresponding author**.

vaseekaranjohn31@gmail.com

Funding: None

Competing Interests: None

Copyright: © 2023 Vaseekaran & Srinivasan. This is an open access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

Abstract

Objectives: To determine the significance of each object (node) in a graph, researchers often employ link analysis techniques, such as the Hyperlink-Induced Topic Search (HITS) algorithm. This will be performed for several reasons, including analyzing the confidentiality of social networks and optimizing search results based on the hierarchical nature of the Internet's interconnections. **Methods:** This work proposes a new version of HITS called the Boundary grading HITS method (BG-HITS). We offer a technique for calculating edge weights that uses just the graph's hub and authority parameters but considers the significance of each edge, its associated relationships and associations, and other relevant qualities such as whether or not they are "organization". **Findings:** Experiments on both simulated and real-world web-graph data demonstrate conclusively that our suggested method, when combined with edge weighting, may mitigate the effects of superfluous edges and nodes on the analysis, yielding more favourable and objective results than the previous HITS approach. **Novelty:** HITS is a method for doing link analysis that treats all edges the same in every calculation, much like nearly all other link analysis algorithms. The novelty of the proposed work is, the value of edges in practice varies from case to case and is influenced by the connections and associations between the two terminals. This has been resolved in the proposed approach.

Keywords: Graph; HyperlinkInduced Topic Search (HITS); Link Analysis; Edge Weight Analysis; Recommendation Model; Points of Interest (POI)

1 Introduction

The modern, information-based culture is replete with big, intricate graphs. A large number of individuals are connected through social networks, and the linking structure of online sites (the "internet graph") is another example of a graph[1]. HITS is a prominent link analysis technique that is used to find relevant and useful nodes (items) in keyword phrase graphs based on a given subject or object. HITS used edges to estimate the weights of each node, and then sorted nodes based on their weights, to obtain nodes with high values. Each node will eventually have a stable state thanks to repeated computations. Also, the process of quantification of worth in the long run. Simply put, HITS is a link analysis method in the influencing relationship, where each connection between the two nodes is seen as a kind of impact from target to origin. Connecting two nodes A and B means that any changes made to node B's content will propagate to node A and might affect its own.

At best, the top-ranking search locations may be precisely determined when these connections are homogenous. On the other hand, search query graphs are typically asymmetrical. Although the individual nodes in a query network may have little or no relationship to one another, the linkages (connections) between them are extremely useful for analysis, as demonstrated by several publications in the literature. On the other hand, HITS analyzes every edge the same, even these crucial ones[2]. We contend that the existing approach in HITS is not the optimum methodology for exploiting edges on a modern complex graph. The only result of this plan was a poor evaluation outcome; leading nodes were not always the best; some were simply worthless, such as spamming or common networks.

To minimize the effects of useless edges, it is ideal to first determine their true worth in the analysis. Given this background, we provide an enhanced version of the HITS algorithm—the Edge weighting HITS algorithm—to address some of its shortcomings. Edge weight is assessed and generated in each iteration, similar to the original approach, to determine which edges are most important to the analysis and how to best handle them. As is well-known, an edge's worth is based on how closely its two termini are related to one another conceptually or semantically. As a result, the weight of invaluable edges connecting nodes with no or weak ties should be lower than the weight of other typical edges connecting nodes with strong ties. Our research indicates that a key factor in an edge's value is its ability to connect entities that share some kind of grouping, such as a common category, a subject on a web graph, or a social group or neighborhood in a social network. An overview of the HITS model recommendation is depicted inFigure 1.

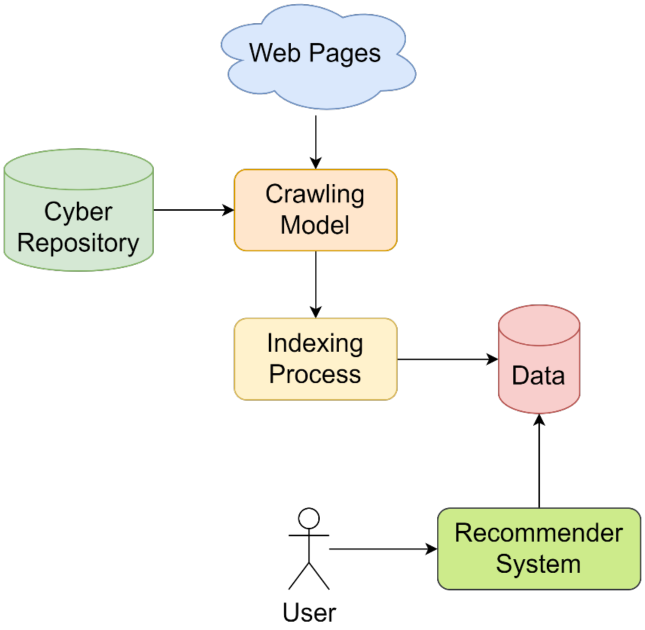

Fig 1. Overview of HITS model recommendation

Checking in at a Point of Interest (POI) is a preference indicator that may be used to compare people's opinions on different places. These geolocation details can be used to provide suggested Points of Interest. We refer to the rich contextual information associated with LBSN check-ins as "impactful characteristics," and we use the example of users' preferences for visiting points of interest (POIs) at different times of day as an illustration of how different users are influenced by these same characteristics because once they perform various activities. David, for instance, would rather hang out at local watering holes on weeknights, while Ela, on the other hand, has different plans for such establishments on Saturdays. When two people have different buying habits, that's another illustration of how people may be different. While one prefers to shop at independent establishments during the week, the other prefers to visit more well-known retailers on the weekends[3].

In most cases, these techniques employ a mixture of attributes. Two approaches might be taken here; one could take chronological and topographical characteristics concurrently, while the other could also factor in the social contacts of the user. However, the sorts of contextual data that a given LBSN has access to are the primary determinant of whether or not the user's preferences are accurately represented. While there are some changes between them, most of the already established systems adhere to the same premise, which is to utilize the Naive Bayes model to compute the chance of a user visiting a POI for each attribute independently and then aggregate the probabilities from all characteristics(4).

Existing methods for providing recommendations can be sorted in several ways, depending on the features they use and the suggestion model they implement. Having commonalities in terms of both location and culture helps bring people together. Companions are more likely to be open to and enthusiastic about exploring new locations and activities together. In an LBSN, these connections take on increased significance for researchers developing recommendation systems(5). Recommendation algorithms, like those utilized in projects where collaborative filing (CF) is the primary approach, benefit from the user-user similarity matrix obtained from social linkages.

User behavior was modeled by Ye et al. visiting behavior by combining both social relationships and geographical elements of his/her check-ins. The term "friend-based CF" refers to the practice of just considering the user's friends when collectively evaluating a POI, rather than all users. They went on to develop a geo-measured friend-based CF (GM-FCF) that works off the premise that people are more likely to spend time with their friends who are physically close by.

All of these researches have one central goal in common: determining an individual user's attraction score to POIs based on her existing network of LBSN acquaintances. For example, in(6), the authors suggest a location-aware recommendation system (LARS) that uses a group's consensus on the popularity of a given point of interest (POI) to account for users' varying travel times and distances. The proximity of each POI to the user's current location is also considered. Users' check-in behavior was modeled by Ying et al. as a two-layer structure composed of a user layer and a location layer, with check-ins serving as the links among the user and points of interest. Considering user choice, the impact of social connections, and the popularity of POIs, they developed a novel random walk, Urban POI-Walk, over this network to score the POIs(7).

Even though these methods are straightforward, which is positive, they have reduced the POI suggestion to a simple item recommendation issue. To determine a user's preference, they simply converted the check-in data together into the user-location matrix, in which each row indicates the number of times a person visited a POI. Yet they are unable to make use of the characteristics that are unique to check-in data, such as location and time. Geographical factors have been found to significantly affect users' navigational patterns. Different strategies have been developed to take advantage of it to more accurately predict the user's tastes. Several of these studies incorporate geographical factors by using a distance range when making location suggestions. The quickest method is to set a minimum acceptable distance(8).

In the aforementioned studies, geographic factors were measured as the average distance involving a user and POIs or between other such locations. Another direction suggests a distance-based model using the distance between a user's visited POIs, on the assumption that this distance follows a certain distribution. The geographical impact, for instance, is represented by supposing that the range between the POIs is distributed according to a power law. When analyzing the distribution of check-ins, Cheng et al. assumed the existence of many central locations. Check-in clustering and spatial preference modeling are employed to determine the locations of these nodes of activity.

A multicenter Gaussian model based on the distances between the POIs. Similar methods are used, but the scale at which they are employed varies. They used a tailored kernel density estimate technique to represent each user's geographical preference, as opposed to the two prior research that assumed POI distances would be evenly distributed among users. In LBSNs, time is a major component that affects the frequency with which people check-in. Strong cyclic patterns may be seen in human migration throughout several periods, from weeks to days(9). For example, whereas one user may like to check in at a dining establishment for lunch on a weekday, another user may have a strong preference for going out for drinks on a Friday night or doing some weekend shopping. Various researchers are making use of temporal characteristics to more accurately predict a user's preferred frequency of visits. To determine the likelihood of a user visiting a certain POI at a given time of day or week, Gao et al. modeled the user's temporal preferences. Users' time preferences were modeled.

By supposing it has a distribution similar to a Gaussian mixture. Using the Bayes rule, they further merged temporal preferences into an individual's and a group's preferences. Two studies, one by Gao et al. and the other by Yuan et al., used a low-rank matrix factorization to make use of temporal features for POI recommendation. To accommodate users' Yuan et al. established many time slots in a day, localized the user's check-ins into some of these time slots, and used a customized form of user-based CF to add the time component for computing the user similarity. A collaborative strategy was presented by Yang et al. to learn a user's temporal preference from other users who have the same preference, taking into consideration the sparsity of the user's check-in data.

By using commonalities between users and across physical locations, Chow and I used a collaborative approach to learning about users' temporal preferences. The primary benefit of this research is that, without segmenting the data, we can construct a posterior distribution to infer a user's temporal preference while visiting a POI. By breaking down a flow of time that is, in theory, continuous into discrete periods. Users of the LBSN have also demonstrated what we term "transitional preferences" about the sequence in which they visit points of interest (POIs). Checking in at a restaurant is one such scenario.

A small number of studies have looked at how users' transitional preference patterns affect their site usage. Liu et al found the transitional favoritism of various POI types. First, users are grouped into subgroups according to the frequency with which they frequent specific categories. By using a cooperative framework, that employs a user-based similarity technique, we can forecast the next POI category a user will visit. By using all of the users' check-in information to build a location-location transition graph (L2TG), Zhang and Chow were able to extract the sequential preferences. Next, they determined the likelihood of a user going to a certain POI by applying an n-order cumulative Markov chain to the whole graph.

The final model for suggesting Points of Interest considers all three types of influence—sequential, social, and geographical—in its calculations. The aforementioned methods were designed to make use of data from a wide range of contexts, including social, geographical, historical, and transitional factors. But they are often designed for certain situations. All nuances, and it's tough to use them in a generic way to manage different types of inputs. However, they're just touching on a few aspects. The POI recommendation method suggested in this article, on the other hand, is a flexible framework that can accommodate a wide range of contextual factors.

To overcome these issues, the proposed system has been designed and the major contributions of this research work are as follows:

- In this paper, we introduce BG-HITS, a technique that improves upon the original HITS algorithm by using boundary grades. Edge weight is evaluated and calculated at each iteration, an improvement over the traditional method, to determine how to best manage the most important edges in the study.
- As is well knowledge, the value of an edge depends on the connections or content similarities between two endpoints. Therefore, the weight of useful edges connecting two nodes with no or minimal link ought to be less than the weight of regular edges connecting two nodes with a tight relationship.
- We find that the group-based characteristic of an edge—a link between items that share the same categorization, subject, social circle, or population on a web graph or social network—is a significant predictor of an edge's worth. The "group-based" feature is considered in the proposed method.
- The work provides the foundation for the assessment and calculation of edges by using this technique, the research is not impacted by insignificant edges, and the node rate may be detected with more accuracy.

2 Methodology

Before delving into the specifics of this topic, we will first provide some fundamental definitions. On the World Wide Web and social networks, information was modeled as a directed graph $G = < V, \ E \ >$, where $V \ = \ 1, \ 2, ..., m$ is the set of nodes, each representing a page or an individual; $E \ = < i, \ j > (i, \ j \ \in V)$is the collection of edges, each representing an URL of the webpage from i to j or an adopt interaction of individuals from i to j. The HITS method gives two numeric qualities to each node, termed authority and hub, with the default value of each edge being 1. We may calculate these node ratings using an iterative process. Let $b \ = < b1, b2, ..., bm >$ and $g \ = < g1, g2, ..., gm >$ represent the authority and hub scores, respectively, of the graph. The computation of HITS can be expressed as follows:

$$g_j = \sum_{j \to l}^{l \in H} b_l \ and \ b_k = \sum_{l \to k}^{l \in H} g_l$$

As previously stated, the HITS approach is unsuitable for analyzing the complex graphs of today since it treats each edge similarly. Therefore, our enhancement of HITS centers on the evaluation of edges and the computation of weights. Typically, the value of an edge is determined by the similarity or reliance of two ends' contents or internal features. However, because this strategy is highly dependent on the unique characteristics of each network, it cannot be used universally for many graph types. The purpose of this research is to determine the value of an edge based just on the hub and permission of two endpoints and to establish a global way to evaluate edges. Our first step is to identify the factors that contribute to the development of a group-based edge into a functional one. The goal is to use this association as a criterion for determining the "group-based" characteristic.

This value is distributed evenly to its edge's incident. Consequently, the authority may be viewed as a metric for measuring the value of its edges, and this value is shared evenly with its incoming edges. In an ideal situation, where nodes belong to the same subject or group, the greatest group-based score of each edge may be treated equally in analysis, and the predictions of each path on the network should be comparable. In a perfect world, the value assigned to an edge by the hub is identical to the value assigned to it by the leadership, as well as the connection between the center and commitment is the sum of these distributive values along a single edge, even though every edge on a graph has two ends across the same collection and the exact significance. The relevant edges of trash nodes are an example of the none organization edge, which connects nodes that do not belong to the same category. We assume that the difference between these given values is much bigger for the non-group-based edge than it is for the group-based edges that join together vertices that all belong to a single grouping. Consequently, the correlative rate, which represents the disparity of the assigned values of a single edge, is defined as follows:

$$js_{xy} = \left(\bar{g}_x - \bar{b}_y\right| = \frac{g_x}{(F\{j \to l\}|} - \frac{b_y}{(F\{l \to k\}|}$$

This correlative frequency is used to quantify and evaluate the ”group-based” quality. We hypothesize the correlation degree of control, the greater its association and ”cluster” feature, as well. To demonstrate this assertion, we conduct trials comparing the combined charges s_{xy} of regular and ”non-group-based” boundaries. Without sacrificing generality, we experiment with link-farm spam as a sample of a ”non-group-based” edge. Here remain dual types of link-farm, namely outbound as well as inbound. Figure 2 shows the proposed model architecture.

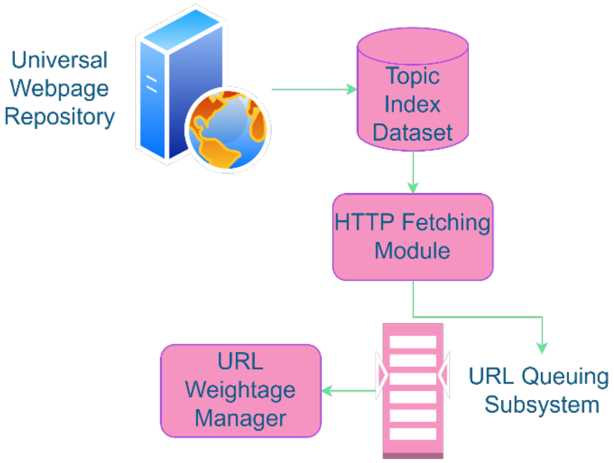

Fig 2. Proposed Model Architecture

Consequently, the terminus of a connection ranch might consume a large in-degree, but the minimal connection to the rest of the graph. In contrast, the outgoing scenario consists of the outgoing control from the junk one and the basis of the outgoing link farm, which would be a large encompassing but have minimal connection to the remainder of the graph. Our investigation compares the correlative rates of edges on sample spamming graphs. Because we cannot evaluate the veracity of edges in a real network, we employ a random display in which all edges are generated at random. The connotation in the arbitrary graph is approximated by an arbitrary function that generates group-based boundaries. Spam is generated by incorporating junk nodes and connection ranches hooked on arbitrary networks as examples of ”non-group-based” edges.

In addition, the correlative rates of edges are provided, with ”group-based” boundaries characterized and ”non-group-based” boundaries signified. The contrast between the correlation rates of normal and ”non-group-based” instances is readily apparent from the above graph. As a result, the correlative frequency of ”pro” edges is very variable and typically appears in the list's top highest values, whereas the correlative rate of normal edges is rather steady and appears in the list's top lowest values. Normal edges have an overall correlative rate of about 0.1175 and a confidence interval of about 0.12, which is near zero, but non-group-based edges have an average correlative rate of about 0.25 with a confidence interval of about 0.13, which is substantially higher than the typical boundary.

The following rule encapsulates this edge property. The greater the edge's correlative rate, the less exact its possible ”non-group-based” weight. The correlation rate of an edge, the more exact the group-based potential of its weight. Further, we are still running trials with different phishing circumstances to help us better understand the characteristics shared by groups. We concentrate on the situation of self-link in phishing nodes. Specifically, in repeatedly posting with self-link, each spam node

Table 1. Algorithm 1: Boundary Grading HITS Model

```
Input: Nodes and Edges as $G \; = < V, \; E \; >$
Steps:
1. Initialize edge weightage x as 1.
2. Estimate $\bar{x} \; and \; \varnothing(x)$.
3. Calculate $-$ and $-$ while $x_{jk} \leq \bar{x} + \varnothing(x)$.
4. Perform the normalization process of authority and Hub space.
5. Separate edge distribution as $\frac{x_{jk}}{\sqrt{\sum_{j \to l}^{l \in H} x_j l^2}}$.
6. Estimate the weightage of the edges as: $-$ while $\nabla x_{jk} = x_{jk} . (g_j - b_k)$
7. Normalize the edge vector estimation as: $x_{jk} = \frac{x_{jk}}{\sqrt{\sqrt{\sum_{j \to l}^{l \in H} x_j l^2} . \sqrt{\sum_{l \to k}^{l \in H} x_l k^2}}}$
8. Return the identified hub and authority list values.
```

contains a self-link. We create a self-link to each spam graph node in the experimental data. Self-link in a garbage node enhances the non-correlation of its ”non-group-based” edge. This is a crucial attribute of the ”group-based” distinctive, utilized to provide the assessment of the edge. Here, we see that a garbage one typically has a wide gap between its center and its control. Through extrapolation, we identify nodes with an exceptionally significant mismatch between hub and authority to be possible garbage nodes.

Following the preceding guideline, we attempt to construct self-links to them to improve the performance of edge weighting. As previously stated, the edge value in the analysis is dependent on group-based characteristics. However, the criteria and measurement of the group-based feature only provide a relative evaluation of the value of the edge and its connection. Then there is the challenge of identifying the true value of an analysis's edge. To address this issue, we suggest adjusting the edge weight in each of the algorithm's iterations based on the correlative frequency and group-based criteria; edge weight will identify the value of an edge in the investigation.

3 Results and Discussion

To assess the effectiveness of the suggested technique, we conduct tests on simulator data sets along with legitimate web data and compare the results of the proposed method with those of the previous system. Our research using random graphs is carried out as follows. On undirected graphs and spamming simulations, we conduct HITS and BG-HITS. We compare separately the lists of nodes arranged by node strength as authoritative and hub. The most weighted nodes in the HITS and BG-HITS results for a random network with 350 members and its phishing simulations with 354 nodes.

We can observe that the results of HITS and BG-HITS do not differ significantly, as the respective lists of nodes with the greatest rate are identical. However, the outcomes of the two strategies are distinct. Following this, the HITS technique places the simulated spam node at the top of the list, but the BG-HITS proposal does not. In reality, the weight of the advertising node in BG-HITS seems to be the smallest value in the list. Specifically, the outcome of BG-HITS in the spamming situation is comparable to the result in the regular case. Experiments on comparable graphs get the same findings as the preceding example.

We observe that, compared to the original HITS approach, the proposed method produces more logical and accurate findings, especially in complex scenarios such as spamming. Regarding actual data examples, we then conduct a similar experiment using the WEBSPAM-UK20064 dataset, which has over 11302 nodes. Nevertheless, a strength of the HITS method is the ability to apply to a data-generated subgraph with a root node, thus we exclusively utilize query graphs built from this dataset. The root node may be erroneously regarded as an outgoing spam node; thus, we also delete the cluster center from these search topologies. In particular, the input graph has 3218 nodes and its root node is www.adslnet365.co.uk. The figure depicts the outcomes of an experiment conducted on a query network with 3218 nodes with a root node. To examine the outcomes of the two algorithms, we employ the spam and normal labels of nodes directly from WEBSPAM-UK2006. There is a distinction between these outcomes.

As a consequence of the HITS algorithm, spam networks have such a high rate that they rank at the top of the list of nodes with the highest rates, however as a result of the BG-HITS method. Here, spam node charges are the deepest standards in the list. Likewise, we do trials on various request grids with the same outcomes. The trials support our hypothesis that the BG-HITS method is capable of handling false/fake factors on graphs. However, we encountered an issue with the execution time of the proposed procedure. Specifically, we compare the execution times of the HITS and BG-HITS algorithms under various input graph scenarios. We can observe that the time complexity of BG-HITS grows considerably more rapidly than that of HITS as the scope of the effort chart grows and that this is a serious issue.

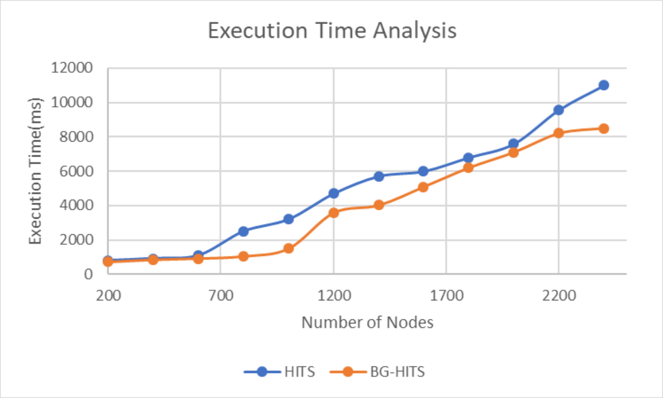

Fig 3. Algorithm Performance Analysis

Three common measures, accuracy, precision, and recall, are adopted to examine the performance of the presented techniques and algorithms. The performance of proposing the next site is initially assessed based on its correctness. If the following POI is in the top(K) suggested POIs, the suggestion is correct (accuracy of 1) We provide the average degree of precision as the proportion of successful suggestions to the overall number of suggestions. By measuring accuracy, we award each user's advice a value of one or zero, regardless of the number of accurately detected or missing POIs. A recommendation model's covering of all visited Data points by a user, in addition to the next visited POIs, is another crucial component. The proposed algorithm performance analysis has been compared with the HITS algorithm in Figure 3.

To determine this, we also examine the measurements of accuracy and recall. To determine the accuracy of our suggestion list, we utilize the ratio of accurately found POIs to suggested POIs. To determine how effectively the recommendation model covers the POIs visited by the user, the ratio of found POIs to those visited either by the user in the test set is calculated. We utilize each user's previous check-in data to develop the model and to simulate future check-ins. Every data set is separated into sets for training and tests for this purpose. The check-in data collected during the first 8 years are utilized to train the recommender system and forecast the testing results. Figure 4 compares Precision, Recall, and Accuracy Analysis.

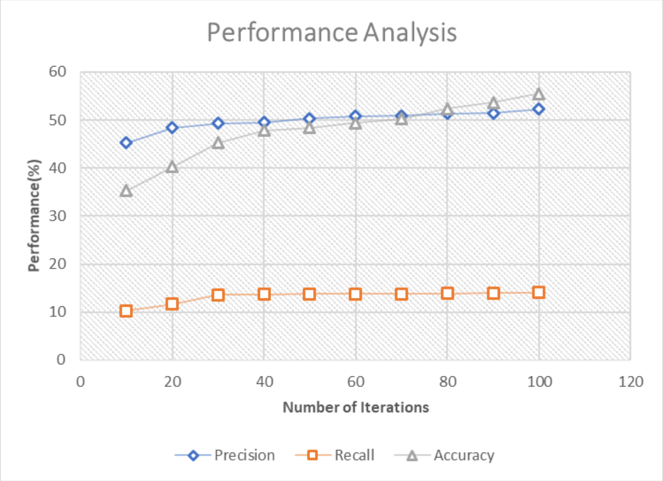

Fig 4. Precision, Recall, and Accuracy Analysis

The algorithm accuracy, recall, and correctness are all examined using the top(K) between 5 and 100. M, the maximum iteration, is set to 50, and h, the number of neighbors, is set to 4 for assessing the BG-HITS. Computer hardware consisting of an Intel Core i5 CPU running at 2 GHz and 4 gigabytes of memory is used for the method validation. The virtual machine is version 10.11.3 of MAC OS X. Python is utilized to implement both the suggested and baseline methods. This section provides an analysis of the evaluation outcomes. This APPR method is associated with existing suggestion points of departure in terms of accuracy, precision, and recall measures. After that, several significant discoveries are discussed. The influence of the number of iterations on the accuracy of BG-HITS recommendations is investigated further. The influence of cluster size on the superiority of the suggestions is subsequently tested.

Furthermore, a comparison is made between HITS and BG-HITS concerning the amount of time required for training using each method. Evaluating our POI suggestion method in comparison to two reference implementations. Characteristics are assumed to be unrelated to the outcome using this methodology. Using this technique, we can separately model how each feature of Xi affects the chance that a given person would visit a certain POI. The full-joint model avoids the shortcomings of the Naive Bayes approach by specifying the relationships between the likelihood of visiting a POI and all relevant attributes Xi. It uses a supervised learning model, just like the one presented in this work, to make its POI recommendations. F1-Score based recommendation value analysis has been listed in Table 2 which compares F1-score values of proposed BG-HITS and existing conventional HITS method.

Table 2. Recommendation Analysis

	F1-Score	
Number of Recommendations	**HITS**	**BG-HITS**
5	0.31	0.47
10	0.35	0.482
20	0.42	0.496
30	0.45	0.525
40	0.47	0.552
50	0.475	0.625
60	0.482	0.655
70	0.489	0.679
80	0.492	0.688
90	0.495	0.712
100	0.499	0.722

As a result, the most important interplays between the features are considered, and the impact of particular components on user activity may be assessed. Unfortunately, the least accurate and exact POIs were returned, and the overwhelming bulk of POIs frequented by the recall's intended audience was not included. Since we're working with limited data, this is because LBSN has a very recent history of available check-ins. The recommended total joint system performed poorly outside of the training set. HITS delivers the highest results when it comes to all three metrics over a spectrum of top(K) values throughout both large datasets by combining the advantages of the Naive Bayes technique with the Full-joint model.

We take into consideration the interplay between characteristics by learning the decision tree from a subset of features. We accommodate for the relative scarcity of data by learning partial models rather than a complete model. It may seem unexpected, as considering additional information is expected to improve the quality of suggestions. In the event of sparse data, a simpler algorithm with fewer characteristics to learn leads to more precise suggestions. We have incorporated the benefits of both methodologies into our recommendation architecture by efficiently utilizing all available data. When compared to the model trained from the New York data set, the model learned from the Tokyo data set yields greater precision and accuracy values and almost the same recall value.

Concerning the number of iterations necessary to achieve peak efficiency, these two data sets exhibit distinct behaviors. The stated discrepancies may be a result of the greater average number of POIs per customer in the Tokyo given dataset as opposed to the York data set, which resulted in greater precision and accuracy as well as a lower number of needed iterations to attain convergence. Consequently, we can infer that the quantity of the check-in history for each user's data set has a considerable impact on both the correctness of the recommendations and the time necessary for the system to converge.

4 Conclusion

In this study, we improve prior HITS algorithms by incorporating a new organization edge weight feature. We explore the logic and suggest a method for calculating the edge's weight based on this association. In this paper, we introduce BG-HITS, a technique that improves upon the original HITS algorithm by using boundary grades. Edge weight is evaluated and calculated at each iteration, an improvement over the traditional method, to determine how to best manage the most important edges in the study. As is well knowledge, the value of an edge depends on the connections or content similarities between two endpoints. Therefore, the weight of useful edges connecting two nodes with no or minimal link ought to be less than the weight of regular edges connecting two nodes with a tight relationship. We find that the group-based characteristic of an edge—a link between items that share the same categorization, subject, social circle, or population on a web graph or social network—is a significant predictor of an edge's worth. The work provides the foundation for the assessment and calculation of edges by using this technique, the research is not impacted by insignificant edges, and the node rate may be detected with more accuracy. Experiments demonstrate that the BG-HITS method can result in superior retrieval performance compared to the original techniques, particularly in complex scenarios such as spamming. From the experimental results, it is found that the proposed work has outperformed in terms of 3977.5ms of execution time, having an average F1-score value of 0.6 and an accuracy of over 90%.

References

1) Petropoulos F, Apiletti D, Assimakopoulos V, Babai MZ, Barrow DK, Taieb SB, et al. Forecasting: theory and practice. *International Journal of Forecasting*. 2022;38(3):705–871. Available from: https://doi.org/10.1016/j.ijforecast.2021.11.001.
2) S, Vaseekaran J, Srinivasan N. Hypertext Induced Topic Search framework based on Relational Semantic Analyzer. *Solid State Technology*. 2020;63(5). Available from: http://solidstatetechnology.us/index.php/JSST/article/view/7914.
3) Dayeh MA, Sartawi B, Salah S. A Bias-Free Time-Aware PageRank Algorithm for Paper Ranking in Dynamic Citation Networks. *Intelligent Information Management*. 2022;14(02):53–70. Available from: https://doi.org/10.4236/iim.2022.142004.
4) Zhou J, Li X, Shang L, Luo L, Zhan K, Hu E, et al. Hyperlink-induced Pre-training for Passage Retrieval in Open-domain Question Answering. 2022. Available from: https://doi.org/10.18653/v1/2022.acl-long.493.
5) Lu Y, Ma K, Duan J. Influence Model of Paper Citation Networks with Integrated PageRank and HITS. *IEEE 24th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*. 2021;p. 1081–1086. Available from: https://doi.org/10.1109/CSCWD49262.2021.9437678.
6) Ito T, Harakawa R, Iwahashi M. Word Clustering Using Graphical Lasso-Guided PCA for Trend Analysis of COVID-19. *IEEE 10th Global Conference on Consumer Electronics (GCCE)*. 2021;p. 200–201. Available from: https://doi.org/10.1109/GCCE53005.2021.9621893.
7) Alghamdi H, Alhaidari F. Extended User Preference Based Weighted Page Ranking Algorithm. *National Computing Colleges Conference (NCCC)*. 2021;p. 1–6. Available from: https://doi.org/10.1109/NCCC49330.2021.9428844.
8) Baker MR, Akcayol MA. A novel web ranking algorithm based on pages multi-attribute. *International Journal of Information Technology*. 2022;14(2):739–749. Available from: https://doi.org/10.1007/s41870-021-00833-5.
9) Du H, Kang YB. An open-source framework for ExpFinder integrating N-gram vector space model and μCO-HITS. 2021. Available from: https://doi.org/10.1016/j.simpa.2021.100069.