A Similarity-Inclusive Link Prediction Based Recommender System Approach

Zuhal Kurt[1, *], Kemal Ozkan[2], Alper Bilge[3], Omer Nezih Gerek[4]
[1]*Department of Mathematics - Computer, Eskisehir Osmangazi University,*
[2]*Department of Computer Engineering, Eskisehir Osmangazi University,*
[1,2]*ESOGU Meselik Yerleskesi, 26480 Eskisehir, Turkey*
[3]*Department of Computer Engineering, Eskisehir Technical University,*
[4]*Department of Electrical and Electronics Engineering, Eskisehir Technical University,*
[3,4]*Eskisehir Teknik Universitesi Iki Eylul Kampusu, 26555 Eskisehir, Turkey*
zkurt@ogu.edu.tr

***Abstract*—Despite being a challenging research field with many unresolved problems, recommender systems are getting more popular in recent years. These systems rely on the personal preferences of users on items given in the form of ratings and return the preferable items based on choices of like-minded users. In this study, a graph-based recommender system using link prediction techniques incorporating similarity metrics is proposed. A graph-based recommender system that has ratings of users on items can be represented as a bipartite graph, where vertices correspond to users and items and edges to ratings. Recommendation generation in a bipartite graph is a link prediction problem. In current literature, modified link prediction approaches are used to distinguish between fundamental relational dualities of like vs. dislike and similar vs. dissimilar. However, the similarity relationship between users/items is mostly disregarded in the complex domain. The proposed model utilizes user-user and item-item cosine similarity value with the relational dualities in order to improve coverage and hits rate of the system by carefully incorporating similarities. On the standard MovieLens Hetrec and MovieLens datasets, the proposed similarity-inclusive link prediction method performed empirically well compared to other methods operating in the complex domain. The experimental results show that the proposed recommender system can be a plausible alternative to overcome the deficiencies in recommender systems.**

***Index Terms*—Bipartite graph; Link prediction; Recommender systems; Similarity.**

I. INTRODUCTION

In recent years, the amount of data that is accessible online has expanded exponentially. Recommendation systems consist of a particular sort of information filtering method that provides recommendations about items based on the interests that a user states. Generally, recommender systems are employed in e-commerce sites and customer-adapted websites. Users demand comfort and convenience in their interactions and the business demands a higher chance of commerce. Hence, the success of the recommendation system is imperative for both users and e-commerce sites. Satisfaction depends on the generation of precise and dependable recommendations. In general, the prediction of ratings for items that have not been considered is achieved by using customer profiles [1]. Depending on the application domain, items can be movies, websites or other products discovered on an online store. For example, Amazon and Netflix use recommendation systems in the sense that Amazon typically suggests books and other articles (as well as many types of commercial items), and Netflix typically suggests movies and TV series to their customers. Even though various algorithms for recommender systems have been developed in recent years, there are still high levels of enthusiasm in this area caused by the growing requirement on functional processes, which can supply customized recommendations and help to deal with information overload problem [1], [2].

Recommender systems are generally categorized according to their approach to prediction of ratings. In general, there exist two primary recommendation methods, i.e., content-based filtering (CBF) and collaborative filtering (CF) methods. Techniques of content-based filtering are usually dependent on the similarity of items to the objects that were previously preferred by the user [3]. On the other hand, CF techniques depend on the ratings provided by users with similar tastes and choices [4]. In any case, methods exhibit particular deficiencies. Predictions of CF recommender systems depend on items formerly rated by other users. Therefore, the performance of a system of CF recommendation is dependent on the degree of accessible rating information. Generally, the user-item preference matrix is highly sparse, which accordingly might lead to inaccurate recommendations [2]. Many different algorithms have been proposed to deal with these drawbacks, e.g., the models based on the user-item interaction graphs are aimed to improve recommendation accuracy [5]–[7]. Two node types exist in a user-item interaction graph as items and users. The recommendation in a user-item interaction graph may be moderated as a sub-problem of link prediction, which is a primary issue attempting to predict the probability of occurrence of a connection between two nodes depending

Manuscript received 17 February, 2019; accepted 6 August, 2019.
This paper was supported by the Technological and Scientific Research Council of Turkey under its TUBITAK 3001 program (project No. 116E284, entitled “Developing Image-based Recommender System”).

on the discovered features and other connections between nodes [8], [9]. In a framework for predicting links, there are symmetrical nodes, which ignore the classification of nodes as the subject (user) and object (item). User-item interaction graphs may also be defined as an adjacency matrix with nodes of users and items, which can be represented as a bipartite graph. These graphs have particular nodes (items and users) and three categories of links (item-item, user-item, and user-user) based on varying endpoint combinations. Currently, the type of user-user or item-item links is labeled to be similar or dissimilar, and the type of links between users and items is labeled as like or dislike [10]–[12]. After such adjustment, it is much more appealing to project links of like or dislike, since only items are suggested to the users.

In this paper, in order to address this task, the proposed model is formulated to depend on the representation of complex numbers with real and imaginary parts in the form. In previous studies, similar or dissimilar links were weighted by real numbers, whereas like or dislike links were weighted by complex numbers [10]. Since a complex number provides a natural algebraic link between real and imaginary values, the problem of recommendation could be considered as a problem of link prediction. With the utilization of the proposed method, other available algorithms of predicting links can still be used by no means of change. The proposed representation's validity and efficiency are assessed by evaluating the performance of the proposed recommendation approach in two real-world datasets.

The rest of the study is organized as follows. Section II and Section III introduce background information related to the proposed recommendation approach. Section IV explains the detailed representation of the proposed recommendation algorithm, and Section V experimentally scrutinizes the proposed recommendation approach in two real-world datasets and provides a discussion on the experimental results. Finally, the obtained results are summarized and concluded on the contributions of the proposed algorithm in Section VI.

II. RELATED WORK

The recommender systems that use CBF methods suggest items to users by analyzing the item descriptions in order to identify, which items a particular user might be interested in. The recommended items from CBF recommender systems are similar in content to the items that the user was previously interested in. Thus, item description and user profiling are the principal concerns of a CBF recommender system [1], [3]. There are many different ways to describe items and users for content-based algorithms [13]. CBF recommender systems usually examine the characteristics of items that were automatically derived by information recovery techniques. However, there are complicated algorithms to tokenize textual documents, while methods of feature extraction can be much more difficult to use for multimedia data or items that have various/heterogeneous characteristics. Some of the main issues concerning CBF techniques are constrained content analysis, the new user problem, and overspecialization [2]. An additional issue of CBF recommender systems is that, firstly, a user has to rate an adequate number of items, then the system can predict recommendations.

Unlike content-based recommender systems, the predictions of CF recommender systems depend on items formerly rated by others [13]. CF methods can recommend items to the users based on similar users' interests or habits, without any need for content information about items. First and foremost, the user ratings on the same item are calculated, then predictions on similar users are made [4]. CF recommender systems have the "new user problem", since the system needs choices of a user in order to provide accurate recommendations to that particular user. Furthermore, they also have a "new item problem", which implies that a new item needs to be rated by an adequate number of users before being suggested precisely by the system. The performance of such a collaborative recommendation system is dependent on the degree of accessible rating information. Generally, the number of ratings acquired is fewer in comparison to the number of ratings that is needed to be recommended. That is to say, the user-item matrix is generally quite sparse, which accordingly causes inaccurate recommendations [2].

Many different CF algorithms have been proposed to overcome these difficulties that are generally categorized into three classes: memory-based, model-based, and hybrid schemes [14], [15]. A content-boosted CF algorithm is proposed to improve recommendation accuracy [16]. Then, the hybrid schemes are constructed to combine the advantages of both CF and CBF techniques [17]. These schemes are focused on the modeling and prediction of transactions/interactions. Modeling users and items in a graph structure is a better way to apply CBF, and CF algorithms in one framework [7], [18]. Several CF heuristic algorithms have examined the structure of user–item interaction graphs to enhance recommendation performance [6], [7]. For example, two-layer graph model in the context of book recommendation is described in [18], where the authors propose a graph-based recommendation approach to integrate the CBF approach along with CF approach in the context of digital libraries by representing books and users as nodes. Learning-based algorithms utilize graphs in building effective personalized recommendation models. Such recommendation methods are generally based on explicit feature extraction, that is difficult to implement to graph-structured data due to the requirements of computational capacity and to design features [5], [6], [19]. In another work, a generic kernel-based machine learning approach of link prediction in bipartite graphs is applied to improve the performance of recommender systems [7]. User-item interaction graph models are also able to improve top-N recommendation performance, which is closely related to the business values in real-world recommender systems [10].

III. BACKGROUND

A recommender system may be represented as a particular graph known as a bipartite graph. A simple directed graph, $G = (V, E)$, comprises of vertices connected by edges. Vertices, V, in a directed network are defined as the

nodes being items and users, while edges, E, represent links between the nodes, i.e., ratings. Let U is the set of users and I is the set of items, respectively. Then, V is the union of all users and items $(V = U \cup I)$ and E is the link set of nodes. The notation of any path is represented as $(a_1, a_2, ..., a_{k+1})$, and the path length is denoted by k, whereas two endpoints are represented as a_1 and a_{k+1} connected by the inner nodes of a_i $(i = 2, 3, ..., k)$. Additionally, k links are observed along this path of $(a_i, a_{i+1}) \in E$, where $\mathrm{i} = 1, 2, ..., \mathrm{k}$. When the path of length corresponds to one (i.e. $k = 1$), it means that there is a link to one of the inner nodes. In the following explanations, $N_u(i)$ is described as the set of items that user u rated and $N_i(u)$ is described as the set of users, who rated items in I, where $N_u(i) = \{i \mid (u,i) \in E, i \in I\}$ and $N_i(u) = \{u \mid (u,i) \in E, u \in U\}$.

If there is a connection between two nodes, there are always two links that connect this node-pair, one in each direction. Then, it is possible to reduce the recommendation effort to predict, whether there will be a link in the graph between a user and a specific item. A prediction, which shows the extent of the relevance of any item to a particular user is calculated by using an algorithm of link prediction in graph-based recommender systems [10]. A useful technique to solve the problem of link prediction is to describe a network in the form of a matrix, where link prediction values are calculated by processing such a matrix. Algebraic graph theory utilize the adjacency matrix $\boldsymbol{A}$, where $\boldsymbol{A}_{ij} = 1$ when (i, j) is an edge and $\boldsymbol{A}_{ij} = 0$ otherwise. For undirected networks, generally, the adjacency matrix $\boldsymbol{A}$ is symmetrical, and its eigenvalue decomposition may be considered as

$$\boldsymbol{A} = \boldsymbol{U}\Lambda\boldsymbol{U}^T, \tag{1}$$

where $\boldsymbol{U}$ is an orthogonal matrix and $\boldsymbol{\Lambda}$ is a diagonal matrix. The logic behind usually considering the adjacency matrix's eigenvalue decomposition is that it is possible to calculate a power of the matrix as

$$\boldsymbol{A}^k = \boldsymbol{U}\Lambda^k\boldsymbol{U}^T, \tag{2}$$

which may be used for expressing link prediction methods like the Neumann kernel, the matrix exponential, triangle closing, and rank reduction. The previous link prediction techniques operated with regard to just one type of nodes. Therefore, these methods need customization before being used in a graph-based recommendation system. Such requirement may be addressed adequately with the integration of the hyperbolic sine function to the system, which is applied to the adjacency matrix of the system. The hyperbolic sine of the adjacency matrix gives the summation of odd components of the exponential of the adjacency matrix

$$sinh(\boldsymbol{A}) = \boldsymbol{A} + (1/6) \times \boldsymbol{A}^3 + (1/120) \times \boldsymbol{A}^5 + ... \tag{3}$$

The other decomposition methods like probabilistic latent semantic analysis or non-negative matrix factorization do not have useful characteristics/features [10].

A. Triangle Closing

Nodes in a user-item bipartite graph may have two types of relationships. First of all, for both user-user and item-item links, there is a similarity factor, $e_{similar}$, between two entities. Then, including user-item links and item-user links, there is a preference, e_{like} and $-e_{like}$, of the user on an item due to the necessity of recognizing the asymmetry between the user and the item. Accordingly, in the case of a link from user u to item i with the weight e_{like}, there is always a reverse link from item i to user u with a weight of $-e_{like}$. In this model, e_{like} and $e_{similar}$ are normalized values just for the weights. The triangle closing rule in this model may be described as shown in Fig. 1.

This rule has two parts: users who have denoted the same interest in shared items may be similar (Fig. 1(a)), similar users will be similarly interested in the same item (Fig. 1(b)), and user similarity is transitive among users (Fig. 1(c)). Likewise, items liked by associated users may be similar (Fig. 1(d)), users are prone to interest in similar items (Fig. 1(e)), and, besides that, item similarity is transitive among items (Fig. 1(f)). These rules are the main ideas of CF from a different viewpoint. Thus, these principles may be mathematically stated as $e_{similar} = -e_{like}^2$, $e_{like} = e_{similar} \times e_{like}$, $e_{similar} = e_{similar}^2$.

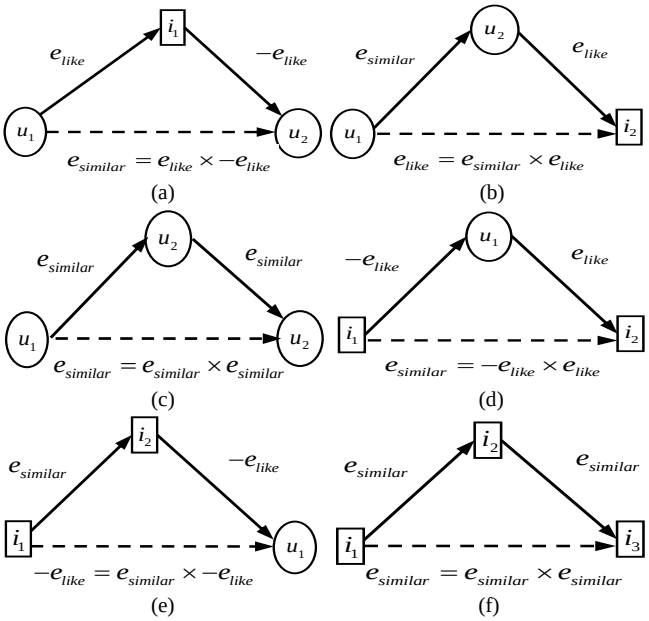

Fig. 1. The triangle closing multiplication rule set: (a), (d) illustrate that the same interest of users/items yields similarity; (b), (e) illustrate that similar users/items will have similar interest; (c), (f) illustrate that user/item similarity is transitive among users/items.

Hence, to solve this system of equations, we need to find two different and nonzero constants, which are $e_{similar}$ and e_{like}. Complex numbers offer an easy way to solve this system of equations, when e_{like} and $e_{similar}$ links are set as $e_{like} = j$ and $e_{similar} = 1$, where j is the imaginary unit. The requirements may be mathematically stated as follows, $1 = -j^2$, $j = 1 \times j$ and $1 = 1^2$. The corresponding

multiplication rules for dislike and dissimilar may then be obtained by multiplying both sides by -1. In another situation, where a user dislikes $(-j)$ an item that is dissimilar (-1) to the one that they are interested in (j) may be expressed as the following equation

$$-j = j \times (-1). \tag{5}$$

In this symbolization, a link has endpoints of the same type, two items or two users must be weighted with a real number. The higher such value, the more similar the endpoints.

On the contrary, a link with an imaginary weight must be an item-user or user-item link based on the sign and interest. For instance, if user u dislikes item i, then the link is weighted with $-j$ from u to i, and the other link is weighted with j from i to u. As opposed to similar links, the dislike and like can only be distinguished when the sign of link's weight and the direction of the link are known at the same time. On the other hand, the value of the weight might define the degree of like or dislike.

B. Adjacency Matrix

The adjacency matrix is described as $\boldsymbol{A} \in \mathbb{R}^{|V| \times |V|}$ given by

$$\boldsymbol{A}(u,i) = \begin{cases} 1, \ if \ (u,i) \in E, \\ 0, \ if \ (u,i) \notin E, \end{cases} \tag{6}$$

when $G = (V, E)$ is denoted as an undirected and unweighted network. The adjacency matrix $\boldsymbol{A}$ is symmetric and square. Therefore, it is possible to derive the number of paths connecting two nodes by calculating the powers of the matrices in unweighted networks. Additionally, it is possible to formulate the number of common neighbours between two nodes u and i $(u, i \in V)$ by taking the square of the adjacency matrix

$$N(u,i) = \boldsymbol{A}^2(u,i), \tag{7}$$

which applies basic triangle closing and may be explained as the number of paths with a length of two among them. This formulization has a significant characteristic: as big as the entry of the square of the adjacency matrix is, these two nodes will be closer. At the same time, the number of paths of any length k from node u to node i can be expressed by the components of $\boldsymbol{A}^k(u,i)$. Therefore, the closeness of the two nodes may be calculated by the weighted sum of powers of the adjacency matrix $\boldsymbol{A}$. Such an example of a link prediction method to unite these results is the matrix exponential

$$exp(\boldsymbol{A}) = \boldsymbol{I} + \boldsymbol{A} + 1/2 \cdot \boldsymbol{A}^2 + \ldots. \tag{8}$$

This function has two main contributions: it considers that all powers of $\boldsymbol{A}$ involve all the paths between two nodes. Also, short paths are prioritized over long paths due to the decreasing weights of the powers. Then, the real numbers are used to represent the user-user and item-item relationships, and the complex numbers are used to express the user-item interactions. The adjacency matrix $\boldsymbol{A}$ of the user-item graph G is defined as follows

$$\boldsymbol{A}(u,i) = \begin{cases} 1 \ if \ u \ similar \ i, \\ -1 \ if \ u \ dissimilar \ i, \\ j \ if \ u \ likes \ i \ or \ i \ dislikes \ u, \\ -j \ if \ u \ dislikes \ i \ or \ i \ likes \ u, \\ 0 \ if \ (u,i) \notin E, \end{cases} \tag{9}$$

where $\boldsymbol{A}(u,i)$ is the value of row u and column i is of the matrix $\boldsymbol{A}$. The matrix $\boldsymbol{A}$ may be conveniently represented as

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{A}_{UU} & \boldsymbol{A}_{UI} \\ \boldsymbol{A}_{IU} & \boldsymbol{A}_{II} \end{bmatrix}, \tag{10}$$

where $\boldsymbol{A}_{II}$ and $\boldsymbol{A}_{UU}$ are the item-item and user-user similarity matrices, $\boldsymbol{A}_{UI}$ and $\boldsymbol{A}_{IU}$ are the user-item preference matrices. Also, the conjugate transpose of $\boldsymbol{A}_{IU}$ can be described as

$$\boldsymbol{A}_{IU} = -\boldsymbol{A}^T_{UI}. \tag{11}$$

The preference matrices are complex matrices, while the similarity matrices are real matrices. In the complex representation-based link prediction method (CORLP) method [10], the authors ignore the relationships between users/items; they represent the bipartite graph as G and the adjacency matrix as $\boldsymbol{A}$ corresponding to

$$\boldsymbol{A} = \begin{bmatrix} 0 & \boldsymbol{A}_{UI} \\ -\boldsymbol{A}^T_{UI} & 0 \end{bmatrix}. \tag{12}$$

Complying with the representation of the adjacency matrix $\boldsymbol{A}$, each entry in the preference matrix $\boldsymbol{A}_{UI}$ has only three different values: $j, -j,$ and 0. Furthermore, $\boldsymbol{B}$, the biadjaceny matrix of bipartite graph corresponding to $\boldsymbol{A}$, is a real matrix. Then $\boldsymbol{A}$ can be expressed as $\begin{bmatrix} 0 & j\boldsymbol{B} \\ -j\boldsymbol{B}^T & 0 \end{bmatrix}$.

Based on the path counting process in the unweighted and undirected networks, the weighted path counting process for paths of length k may be similarly derived by $\boldsymbol{A}^k$. When the relationships between users and items are isolatedly considered, the k^{th} power of the adjacency matrix may be further formulated mathematically as

$$\boldsymbol{A}(u,i) = \begin{cases} \begin{bmatrix} \left(\boldsymbol{B}\boldsymbol{B}^T\right)^n & 0 \\ 0 & \left(\boldsymbol{B}^T\boldsymbol{B}\right)^n \end{bmatrix}, where \ k = 2n, \\ j \times \begin{bmatrix} 0 & \left(\boldsymbol{B}\boldsymbol{B}^T\right)^n \boldsymbol{B} \\ -\left(\boldsymbol{B}^T\boldsymbol{B}\right)^n \boldsymbol{B}^T & 0 \end{bmatrix}, where \ k = 2n+1. \end{cases} \tag{13}$$

Thus, any sum of the powers of the adjacency matrix $\boldsymbol{A}$ may be divided into components that are even and odd, but only the odd components are effective for final

recommendation. Hence, the predictions may be generally applied to $\boldsymbol{A}$ giving

$$P(\boldsymbol{A}) = \lambda \cdot \boldsymbol{A} + \lambda_3 \cdot \boldsymbol{A}^3 + \lambda_5 \cdot \boldsymbol{A}^5 + \lambda_7 \cdot \boldsymbol{A}^7 + \lambda_9 \cdot \boldsymbol{A}^9 + \ldots \quad (14)$$

to guarantee that shorter paths yield more to the predictions, $\{\lambda_1, \lambda_2, \lambda_3, ...\}$ is a decreasingly weighted sequence.

The proposed algorithm similarity-inclusive link prediction method (SIMLP) differs slightly from CORLP method [10] in the modeling of the adjacency matrix and, while calculating the powers of the adjacency matrix and yielding the final recommendation, are in the same procedure. The definitions of user-user and item-item cosine similarity matrix of the preference matrices are available in [20]. Following the combination of these matrices, the main adjacency matrix is built as in (15). Moreover, this adjacency matrix is a square matrix. Hence, the eigenvalue decomposition can be used on this adjacency matrix in (10), (15)

$$\boldsymbol{A} = \begin{pmatrix} u_{11} & \cdots & u_{1n} & r_{11} & \cdots & r_{1n} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ u_{m1} & \cdots & u_{mn} & r_{m1} & \cdots & r_{mn} \\ -r_{11} & \cdots & -r_{1n} & i_{11} & \cdots & i_{1n} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ -r_{m1} & \cdots & -r_{mn} & i_{m1} & \cdots & i_{mn} \end{pmatrix}, \quad (15)$$

where u_{ij} denotes the cosine similarity between the i^{th} and j^{th} users, i_{ij} denotes the cosine similarity between the i^{th} and j^{th} items, r_{ij} expresses the like/dislike relationship between the i^{th} user and j^{th} item, and $-r_{ij}$ expresses the like/dislike relationship between the i^{th} user and j^{th} item in (15).

In our proposed method with another approachment, the link prediction function is multiplied with a parameter α, then the prediction function that is applied to adjacency matrix $\boldsymbol{A}$ is represented as

$$P(\alpha\boldsymbol{A}) = \lambda \times \alpha\boldsymbol{A} + \lambda_3 \times (\alpha\boldsymbol{A})^3 + \lambda_5 \times (\alpha\boldsymbol{A})^5 + \lambda_7 \times (\alpha\boldsymbol{A})^7 + \ldots . \quad (16)$$

IV. RECOMMENDATION METHODOLOGY

Since the closeness values among the nodes are measured by the power sum of the adjacency matrix, the summation of each entry of the top-right and top-left components expresses the degree of whichever item is relevant to a specific user. After summation of these components, the prediction scores that denote item recommendation to a particular user are obtained. These scores are sorted in descending order; thus, the user will like the item if the score is positive or dislike otherwise. Hence, the items with positive and higher values will be recommended to a particular user, if these recommended items are unnoticed by that user. Moreover, top-N recommendation lists are generated for each user by these sorted prediction scores [20].

The testing methodology adopted in this study is the same as in a previous study [10]. The ratings are split by two subsets that are named by training and test sets for each dataset. The test set includes only 5-star ratings and only items that are relevant to the corresponding users. The detailed procedure used to generate the training set and the test set may be defined as follows. Firstly, 10 % of items rated by each user are selected randomly to create a temporary test set, while the temporary training set includes other ratings. After the selection, the 5-star ratings in the temporary test set are further filtered out for the final test set, and the remaining ratings in the temporary test set are combined into the temporary training set for the final training set. Then, the training set is utilized to predict ratings or recommendation scores for each item-user pair.

Nevertheless, rating conversion is necessary for the adjacency matrix's generation of our proposed method, where the ratings in the training set are converted to j or $-j$ based on whether the rating is greater than or equal to 3. Accordingly, in case that the rating is less than 3, it is changed by $-j$, which means that the user states "dislike" for the item; equivalently, when the rating is greater than or equal to 3, j is given to defining "like". Furthermore, if the (u, i) pair is not included in the training set, the corresponding component of the adjacency matrix becomes zero. The rating threshold value is chosen 2.5 for the Hetrec dataset, since this dataset includes decimal rating numbers. By this partitioning process of the dataset, computing the recommendation error becomes less meaningful. Hence, this study is focused on how many relevant items in the test set can be recommended to users. Also, the overall ratio of the items that recommended to all users is calculated. Thus, the performance of the comparison methods is measured by using the metrics, hits rate, and coverage [10], [21], [22]. In the case of the top-N recommendations, the overall hits rate and coverage are described by averaging all test cases:

$$hits\ rate(N) = \frac{\#hits}{|T|}, \quad (17)$$

$$coverage(N) = \frac{|\cup recommend(N, u)|}{\#items}. \quad (18)$$

When the item i is included in the user's u top-N recommendations list for each pair (u, i) in the test set, it will get one hit. The overall hit is symbolized as *#hits*, and the number of test pairs is denoted as $|T|$. Hence, the hits rate can be accepted as the capability to recommend relevant items to users - the recommendation set to user u is denoted as $recommend(N, u)$. Thus, coverage is equal to the percentage of items that the system can recommend. Generally, coverage is utilized to determine models, which recommend a limited number of items, but have a high accuracy. The higher coverage value is not only desirable, but useful to trust the accuracy of the metric results better also [23]. The algorithm performs better when the values of these two metrics are higher.

V. EXPERIMENTAL RESULTS AND DATASETS

The proposed algorithm and other comparison methods

are implemented on two real-world datasets: MovieLens [24] and MovieLens Hetrec [25]. These datasets are publicly stored movie rating datasets that were compiled by GroupLens research from the MovieLens and hetrec2011 websites. The former consists of 100,000 ratings ranging from 1 to 5 from 943 users on 1,682 movies. The MovieLens Hetrec dataset consists of 855,598 ratings ranging from 1 to 5 from 2,113 users on 10,197 movies. Firstly, ratings in these datasets are converted into complex numbers, then the complex biadjacency matrices of these datasets are obtained. Secondly, the cosine similarity measurement is applied to user-item rating matrices of these datasets. Lastly, the user-user cosine similarity matrices and item-item cosine similarity matrices of rating matrices of these datasets are obtained. After combining all these matrices, the main adjacency matrices are constructed as a square matrix for these two datasets as in (10). Therefore, the hyperbolic sine function is applied on the adjacency matrix as a link prediction function [10]. Hyperbolic sine function calculates the sum of the odd powers and gives the shortest path of lengths in bipartite systems. Such function provides a higher score when more paths are connecting two nodes. Therefore, it is needed to have higher powers of the adjacency matrix.

The more paths between two nodes and the shorter these paths are, the most substantial relationship between these two nodes will be in the forecast. Thus, the first experiment was designed to test the performances of the recommendation algorithms based on the link prediction approach with different path lengths for the recommendation. The shortest path of lengths 3, 5, 7, and 9 are found because the sum of the odd powers of bipartite graphs is vital to make a recommendation. For instance, when the path length is chosen as 3, the number of positive value paths with length 3 from user u to item i is more than other path lengths. Hence, if there exist more positive paths from u to i and less negative paths between them, the most probable is that i will be recommended to u. Note that the length needs to be odd and not smaller than 3. As a similar consequence, results of the SIMLP method with top-N recommendations are given. Figure 2 shows the results of the SIMLP algorithm with lengths 3, 5, 7, and 9.

Figure 2 illustrates that the coverage and hits rate decrease as the path length increases in these datasets. Moreover, the proposed algorithm performs much better in the MovieLens dataset than in the Hetrec dataset, since the latter is much sparser and its links between users and items are scarce compared to MovieLens. It still shows a higher performance with length 3 for recommendations in the MovieLens and Hetrec datasets. An item-based top-N recommendation algorithm is used to make a performance assessment. The length of top-N item recommendation lists is increased from 10 to 100. Then, these results are compared with CORLP method based on the fundamental link prediction approach with complex numbers introduced in [10]. Figure 3 illustrates the comparison of results with the CORLP method with the different top-N recommendation. The figure shows that the hits rate of the SIMLP method is higher than of the CORLP method, but the coverage is relatively the same as with the CORLP on the two datasets. Therefore, the link prediction function is modified by scaling with the parameter α as in (16). It can be seen from the obtained results that the modified link prediction enhances the performance of recommendation.

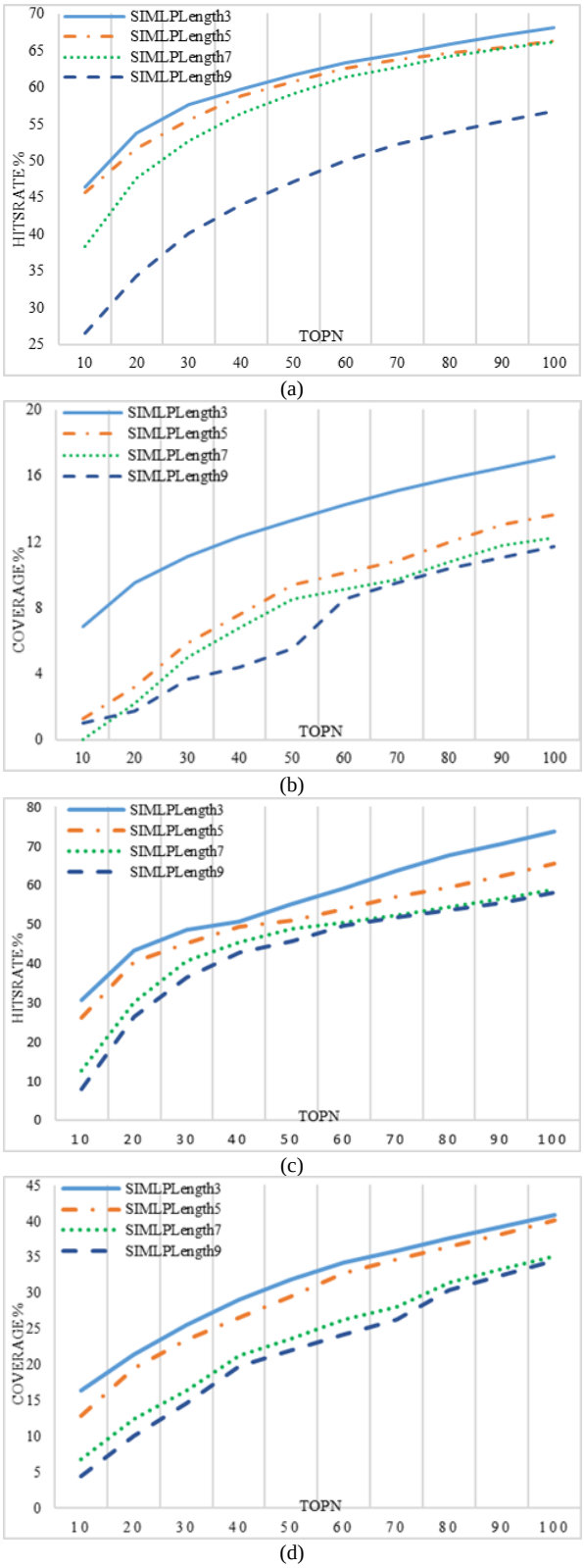

Fig. 2. The coverage (%) and hits rate (%) comparison of SIMLP with different path lengths for top-N recommendation on Hetrec (a), (b) and MovieLens (c), (d) datasets.

The CORLP and SIMLP algorithms are also modified by scaling with a parameter α [26]. While SIMLP can obtain

higher performance with all path lengths, CORLP method performs well only with a path of length 3. Thus, only path length 3 and top60/top100 recommendation lists are considered in experiments, which compare the proposed method to CORLP. Figure 4 illustrates the comparison of hits rate and coverage with the recommendation method CORLP. The results show that the SIMLP achieves higher hits rate and it provides relatively high coverage with decreasing multiplication parameter.

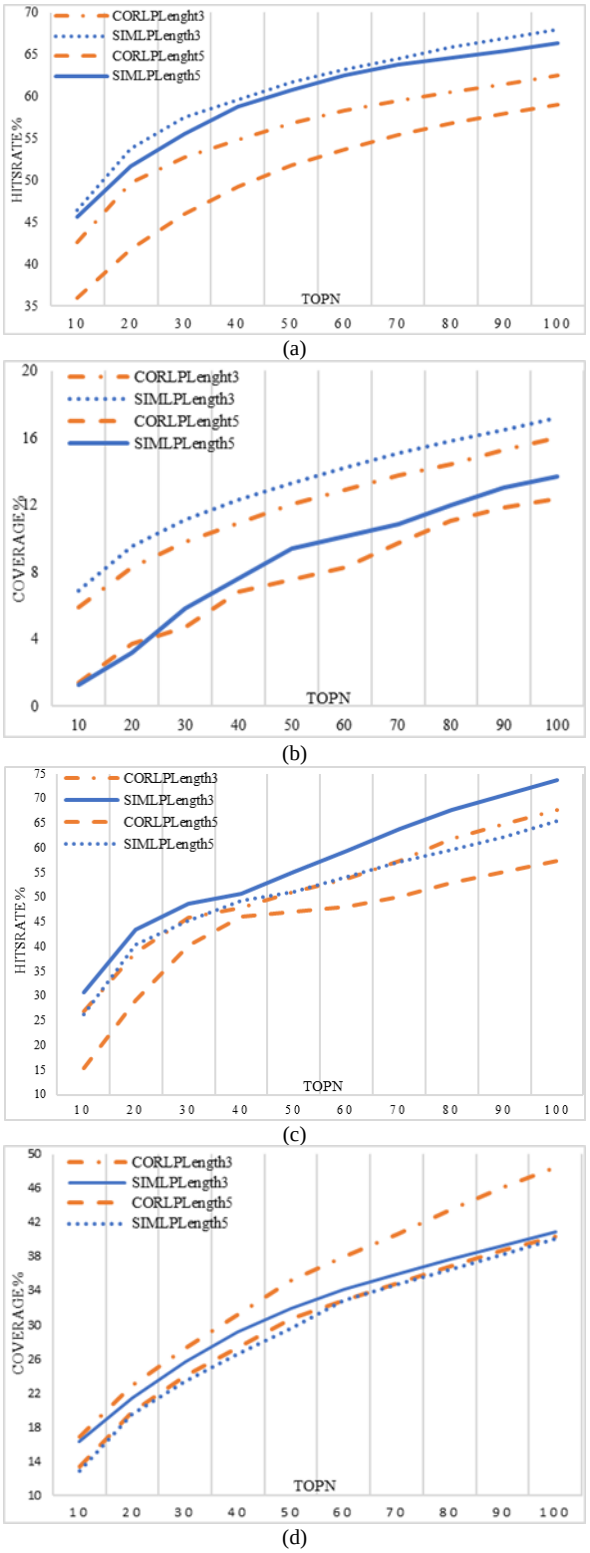

Fig. 3. The coverage (%) and hits rate (%) comparison of SIMLP and CORLP with path lengths 3 and 5 for top-N recommendation on Hetrec (a), (b) and MovieLens (c), (d) datasets.

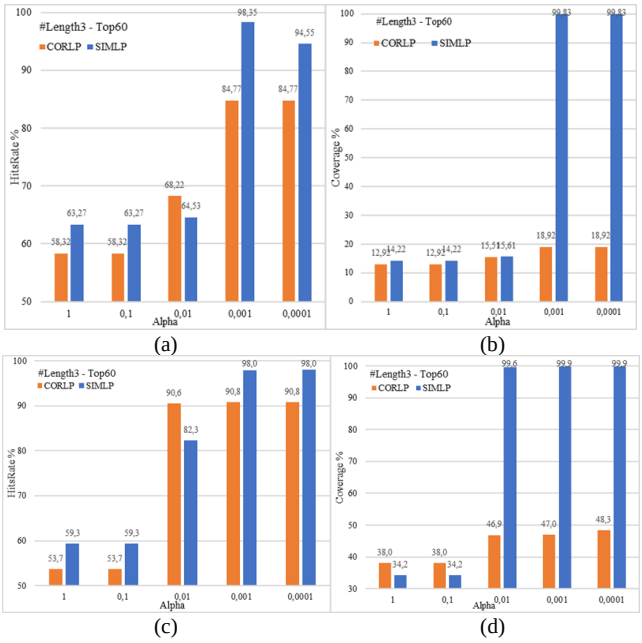

Fig. 4. The coverage (%) and hits rate (%) comparison of SIMLP and CORLP with path length 3 for top60 recommendation on Hetrec (a), (b) and MovieLens (c), (d) datasets.

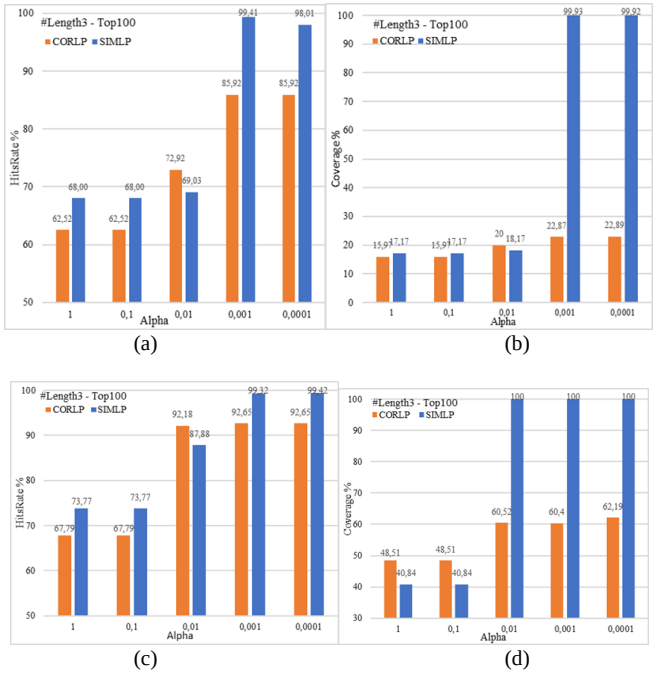

Fig. 5. The coverage (%) and hits rate (%) comparison of SIMLP and CORLP with path length 3 for top100 recommendation on Hetrec (a), (b) and MovieLens (c), (d) datasets.

VI. CONCLUSIONS

Recommender systems are promising technologies to cope with the information overload problem of modern times. Due to the challenging problem of predicting inclinations of individuals based on their limited past preference history, researchers are implementing new strategies to estimate original items to recommend. Graph-based recommender systems are one of such approaches to model relations among users and items in a graph structure and estimate referrals using link prediction algorithms. It is known that complex number-based link prediction approaches, CORLP and the proposed SIMLP methods, obtain higher accuracy

compared to SVD++, ItemBasedPear, Popular and SlopeOne methods in graph-based recommender systems [10]. The proposed recommendation method, SIMLP, is based on such a link prediction approach with the weights in the graph represented by complex numbers that can accurately differentiate *"similarity"* between two users (or two items) and the *"like"* from a user to an item. The experimental results demonstrate that the proposed similarity-inclusive link prediction method performs better than remaining complex number-based algorithms, such as CORLP, regarding coverage and hits rate on the MovieLens Hetrec and MovieLens datasets. Obtained improvements of SIMLP are attributed to the inclusion of similarity factors among users and items. The results indicate that the hits rate of the similarity-inclusive link prediction method is significantly (about 7 %) better than that of other methods in graph-based recommendation systems, whereas the coverage is marginally higher compared to the existing approaches. With the modification of the link prediction function by a scaling parameter, the proposed SIMLP method achieves higher hits rates. The proposed method provides relatively higher coverage at smaller scale parameters, meaning that the cold-start problem of the recommender systems can be easily overcome. Finally, it is concluded that the proposed method deals well with the deficiencies in graph-based recommender systems making the proposed recommender system a preferable alternative.

The design of a graph image-based recommender system, which is based on semantic relationships among images, is considered to be a follow-up of this study.

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest.

REFERENCES

[1] G. Susan, M. Speretta, A. Chandramouli, and A. Micarelli, "User Profiles for Personalized Information Access", in *The Adaptive Web: Methods and Strategies of Web Personalization*, Springer, Berlin, Heidelberg, 2007, pp.54–89. DOI: 10.1007/978-3-540-72079-9_2.

[2] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions", *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 6, pp. 734–749, 2005. DOI: 10.1109/TKDE.2005.99.

[3] M. J. Pazzani and D. Billsus, "Content-Based Recommendation Systems", in *The Adaptive Web: Methods and Strategies of Web Personalization*, Springer, Berlin, Heidelberg, 2007, pp. 325–341. DOI: 10.1007/978-3-540-72079-9_10.

[4] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen, "Collaborative Filtering Recommender Systems", in *The Adaptive Web: Methods and Strategies of Web Personalization*, Springer, Berlin, Heidelberg, 2007, pp. 291–324. DOI: 10.1007/978-3-540-72079-9_9.

[5] Z. Huang, D. Zeng, and H. Chen, "A comparison of collaborative-filtering recommendation algorithms for e-commerce", *IEEE Intell. Syst.*, vol. 22, pp. 68–78, 2007. DOI: 10.1109/MIS.2007.4338497.

[6] T. Zhou, J. Ren, M. Medo, and Y.-C. Zhang, "Bipartite network projection and personal recommendation", *Phys. Rev.*, vol. 76, no. 4, pp. 46–115, 2007. DOI: /10.1103/PhysRevE.76.046115.

[7] X. Li and H. Chen, "Recommendation as link prediction in bipartite graphs: A graph kernel-based machine learning approach", *Decis. Support Syst.*, vol. 54, no. 2, pp. 880–890, 2013. DOI: 10.1016/j.dss.2012.09.019.

[8] L. Getoor and C. P. Diehl, "Link mining: A survey", *ACM SIGKDD Explor. Newslett.*, vol. 7, no. 2, pp. 3–12, 2005. DOI: 10.1145/1117454.1117456.

[9] D. Liben-Nowell and J. Kleinberg, "The link-prediction problem for social networks", *J. Am. Soc. Inf. Sci. Technol*., vol. 58, no. 7, pp. 1019–1031, 2007. DOI: 10.1145/956863.956972.

[10] F. Xie, Z. Chen, J. Shang, X. Feng, and J. Li, "A link prediction approach for item recommendation with complex number", *Knowl. Based Syst.*, vol. 81, pp. 148–158, 2015. DOI: 10.1016/j.knosys.2015.02.013.

[11] J. Kunegis, E. W. De Luca, and S. Albayrak, "The link prediction problem in bipartite networks", in *Proc. of Int. Conf. on Information Processing and Management of Uncertainty Knowledge-Based Systems (IPMU 2010)*, 2010, pp. 380–389. DOI: 10.1007/978-3-642-14049-5_39.

[12] C. Dai, L. Chen, B. Li, and Y. Li, "Link prediction in multi-relational networks based on relational similarity", *Inform. Sciences*, vol. 394–395, pp. 198–216, 2017. DOI: 10.1016/j.ins.2017.02.003.

[13] M. J. Pazzani, "A framework for collaborative, content-based and demographic filtering", *Artif. Intell. Rev.*, vol. 13, no. 5–6, pp. 393–408, 1999. DOI: 10.1023/A:1006544522159.

[14] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, "Evaluating collaborative filtering recommender systems", *ACM T. Inform. Syst.*, vol. 22, no. 1, pp. 5–53, 2004. DOI: 10.1145/963770.963772.

[15] M. Y. H. Al-Shamri and K. K. Bharadwaj, "Fuzzy-genetic approach to recommender systems based on a novel hybrid user model", *Expert Syst. Appl.*, vol. 35, no. 3, pp. 1386–1399, 2008. DOI: 10.1016/j.eswa.2007.08.016.

[16] P. Melville, R. J. Mooney, and R. Nagarajan, "Content-boosted collaborative filtering for improved recommendations", in *Proc. of 18th National Conf. on Artif. Intell.*, Alberta, Canada, 2002, pp. 187–192.

[17] R. Burke, "Hybrid recommender systems: Survey and experiments", *User Model. User-adapt. Interact.*, vol. 12, no. 4, pp. 331–370, 2002. DOI: 10.1023/A:1021240730564.

[18] Z. Huang, W. Chung, and H. Chen, "A graph model for E-commerce recommender systems", *J. Am. Soc. Inf. Sci. Technol.*, vol. 55, pp. 259–274, 2004. DOI: 10.1002/asi.10372.

[19] Z. Huang, D. Zeng, and H. Chen, "A unified recommendation framework based on probabilistic relational models", in *Proc. of 4th Annu. Workshop on Inf. Technol. and Syst.*, 2005. DOI: 10.2139/ssrn.906513.

[20] P. Bedi, A. Gautam, S. Bansal, and D. Bhatia, "Weighted bipartite graph model for recommender system using entropy based similarity measure", in *Proc. of International Symposium on Intelligent Systems Technologies and Applications (ISTA 2017)*, 2017, pp. 163–173. DOI: 10.1007/978-3-319-68385-0_14.

[21] P. Cremonesi, Y. Koren, and R. Turrin, "Performance of recommender algorithms on top-n recommendation tasks", in *Proc. of 4th ACM Conf. RecSys.*, Barcelona, 2010, pp. 39–46. DOI: 10.1145/1864708.1864721.

[22] F. Gedikli and D. Jannach, "Recommendation based on rating frequencies", in *Proc. of 4th ACM Conf. RecSys.*, Barcelona, 2010, pp. 26–30. DOI: 10.1145/1864708.1864755.

[23] F. Cacheda, V. Carneiro, D. Fernández, and V. Formoso, "Comparison of collaborative filtering algorithms: Limitations of current techniques and proposals for scalable, high-performance recommender systems", *ACM Trans. Web*, vol. 5, no. 1, pp. 1–33, 2011. DOI: 10.1145/1921591.1921593.

[24] F. M. Harper, and J. A. Konstan, "The movielens datasets: History and context", *ACM Trans. Interact. Intell. Syst.* vol. 5, no. 4, pp. 1–19, 2016. DOI=http://dx.doi.org/10.1145/2827872.

[25] GroupLens Research Group, "Information Heterogeneity and Fusion in Recommender Systems (HetRec 2011)", May 2011. [Online]. Available: http://ir.ii.uam.es/hetrec2011/datasets.html.

[26] M. Shimbo and T. Ito, "Kernels as link analysis measures", in *Mining Graph Data*, John Wiley & Sons, 2006, pp. 283–310, Ch. 12. DOI: 10.1002/9780470073049.ch12.