Int. J. Nonlinear Anal. Appl. 17 (2026) 3, 43–54
ISSN: 2008-6822 (electronic)
http://dx.doi.org/10.22075/ijnaa.2024.34433.5144

A hybrid cumulative knowledge framework for friend recommendation in social networks

Mahdi Bazargani[a,*], Nafiseh Fareghzadeh[b], Mehdi Afzali[a], Shiva Karimi[a]

[a]*Department of Computer Engineering, Zanjan Branch, Islamic Azad University, Zanjan, Iran*

[b]*Department of Computer Engineering, Khodabandeh Branch, Islamic Azad University, Zanjan, Iran*

(Communicated by Seyyed Mohammad Reza Hashemi)

Abstract

Recently, recommender systems have been a vital part of social network applications to help users make the best choices and expand their heuristic experiences. Friend recommender is a popular service that has been widely developed and focuses mainly on relationships and user interests. Traditional friend recommender systems suffer from some shortcomings that hamper their effectiveness. Hybrid techniques to recommender systems can significantly improve the overall quality of related applications. In this research, a hybrid cumulative knowledge framework (HCKF) has been designed for friend recommendation in social networks. HCKF performs a huge amount of precise evaluations on data, including pre-processing on the data, data preparation, optimal features selection and user clustering in the background; therefore, friend suggestions can be provided at an acceptable performance, precision and speed. HCKF solves the cold start challenge of new users, increases the accuracy of providing attractive friends to users and also minimizes the error rate. The experimental results demonstrate the advantages and effectiveness of the proposed framework in social networks. After simulating HCKF and calculating the error rate and accuracy, the final precision rate of the friend suggestion to the newly logged-in user is equal to 90%. Moreover, the average accuracy in the proposed framework is equal to 0.0197, which has improved significantly compared to other related methods.

Keywords: Friend Recommender, Hybrid Framework, Social Networks, Cumulative Learning, New Knowledge
2020 MSC: 00A06, 93C30, 97M70

1 Introduction

In the recent intelligent information society, the development and progress of social networks are taking place on a huge scale. Now, online social networks have become a natural strategy for users to socialize and search for desired information. Mobile technologies made these activities pervasive and accessible to any user. With the emergence of social networks, recommender systems have gained more development momentum. Suggestions generated by these systems serve the purpose of making effective proposals about items of information or services, based on the end user interests. In general, recommendation systems use user feedback, analyze and process user preferences and make the desired output [7].

*Corresponding author
Email addresses: mbzir@iauz.ac.ir (Mahdi Bazargani), n.fareghzadeh@iauz.ac.ir (Nafiseh Fareghzadeh), afzali@iauz.ac.ir (Mehdi Afzali), shikrm@iauz.ac.ir (Shiva Karimi)

Received: June 2024 *Accepted:* February 2025

Friend recommendation systems that suggest friends in social networks have become useful tools because they encourage users to use new links, thus increasing their interaction and experience. Such recommendations are popular services in social networks, which mainly focus on users' relationships and interests and have been widely studied in recent years. These services suggest new friends so that users can expand their communication and interactions.

These systems are often combined with other technologies and are common in social communications, banking applications, insurance services and financial transactions [2]. In a basic approach, these systems can use the topological structures of a social network to provide suggestions. In this approach, the recommendation is based solely on how people are currently communicating. In another approach, the recommendation can include non-topological information such as user profile, education and professional background. While this method can improve recommendation performance, it largely focuses on static information [8].

Despite these advances, friend recommender systems currently face complex challenges. On the other side, the effective performance of the recommender systems is very important for users in different fields, and there is a need for appropriate approaches and tools to evaluate and measure performance and solve existing challenges. Another fundamental challenge in recommender systems is the cold start problem. This problem occurs when a new user enters into the system, and because there is not enough information about this user, the system will not be able to provide suitable suggestions. Due to the importance of the issue, solving this problem in social networks has been given a lot of attention, and various methods have been presented to improve it. Therefore, it seems very important and necessary to provide a suitable solution that can suggest friends to users without having information about their activity history and functionality in the social network system [24, 20]. Also, another disadvantage of many recent proposals is that many existing solutions ignore valuable detailed information, such as user profiles and special interests [5].

Motivated by existing challenges and the necessity of research, in this paper, a hybrid cumulative knowledge framework, named HCKF, is designed for recommending friends in social networks, in which various and useful techniques, resources and solutions have been developed, based on new knowledge and cumulative learning. HCKF has been designed based on the demographic information of new users and shared information of other users in the social network. The proposed framework solves the cold start challenge of new users, increases the precision and accuracy of providing attractive friends to users and minimizes the error rate. The major innovations and achievements of the proposed framework are summarized as follows:

- Suggest hybrid similarity evaluation measures to find attractive users with maximum depth in social network.
- Effective integration of the content-oriented and knowledge-oriented recommender systems.
- Using an efficient link prediction system based on cumulative learning in the current design.
- Designing an HCKF as a hybrid recommender framework, based on the new knowledge and cumulative learning system to introduce friends in social networks.
- Improving the precision and accuracy of friends' suggestions in the social network using the proposed cumulative learning system and minimizing the error rate.
- Solve the cold start challenge of new users with HCKF.

It is worth mentioning, HCKF is based on the demographic information of new users and shared information of other users in the social network. An important advantage of HCKF is that the mentioned operations are performed in the background of the system, and therefore, suggestions are provided more quickly and accurately. The proposed framework has been evaluated with relevant simulation tools. From the experimental results, it can be seen that, for some common key metrics, HCKF performs better and more effectively than other comparable systems.

The rest of this paper is structured as follows. In section 2, we first review the current status of recommender systems, applications, challenges, and prospects and also review related research literature. In Section 3, the proposed framework and relevant details are explained. Section 4 describes the visualization design and implementation of the HCKF. In section 5, the discussion and comparison of the proposed approach with related work have been done. Finally, the paper concludes with a discussion on the findings and future research directions.

2 Background and Related Work

Basically, the recommender system is a subset of the information filtering structures that seek to predict the score or priority that the active user gives to an information item. The friend recommendation system identifies the behavior of users found in the dataset and provides friend suggestions for the users. In recent years, recommender systems have become very common. Suggestions provided by these systems can generally help the user in making a decision or increase the user's awareness in the field of interest. Recommender systems are useful and provide benefits to both sides of the commercial or non-commercial interactions. This topic has been widely studied in recent years, which mainly focuses on social relationships and special user interests. Recommender systems have different applications and types. Former work distinguishes recommendation techniques into following classes [2, 6, 10]:

- **Content-based filtering:** This methods match the preferences or profiles derived from the attribute information of users, such as their self-description, demographic data and meta-data of items. A user's profile is represented as a vector of key terms, and similarity among user profiles can be calculated with machine-learning or information-retrieval techniques. The limitations of this approach are twofold: first, it only can deal with textual information and cannot work with non-text items such as music or images. Second, it ignores the information of social actions between users, such as the common interests of people from the same community.
- **Demographic Filtering:** In demographic filtering, recommendations are established on a demographic profile of the user. Here recommendation is based on the information provided by users, and it is considered to be similar according to demographic parameters such as nationality, age, gender, etc.
- **Knowledge-based filtering:** In this type of recommender system, the raw material used to generate a list of suggestions is the system's knowledge about the customer and the product. Knowledge-based systems use different methods that can be used for knowledge analysis and include common methods in genetic algorithms, fuzzy algorithms, neural networks, decision trees, example-oriented reasoning, etc.
- **Collaborative filtering:** It makes automatic predictions about the interests of a user by collecting taste information from many users. One challenge of the collaborative filtering approaches is the cold-start problem, which refers to the requirement for initial information of user ratings to make recommendations effectively. Moreover, pure rating-based collaborative filtering approaches can not take advantage of data that may be available in addition to ratings. To address these shortcomings, hybrid recommender systems were introduced, which combine different techniques and can improve performance [15].
- **Hybrid filtering:** Hybrid filtering is a combination of more than one filtering approach. The hybrid filtering approach is introduced to overcome some common problems that are associated with the above filtering approaches, such as the cold start problem, overspecialization problem and sparsity problem. The designers of these types of systems often combine two or more types of the mentioned techniques to increase the performance and reduce the effect of the weak points that those systems have when used alone [14].

In this research, a hybrid friend recommender framework has been designed based on the new knowledge and cumulative learning. From the perspective of related work, in [1], Bazargani and his colleague investigate the problem of cold start users by presenting a recommendation model based on a deep neural network and considering the problem of improving the internet marketing strategy. Moreover, the processing time of the proposed method is less than other related work. In 2017, Zhang and his colleagues presented an efficient incremental dynamic link prediction method in the social network [23]. They conducted their experiments on Facebook and YouTube datasets. They also used the cosine similarity criteria. In their research, they used the criteria of execution time, accuracy to evaluate the performance of the proposed method. In 2018 and the article [4], Chen and his colleagues presented a communication detection method using link prediction in multi-connection social networks. They performed their experiments on YouTube, disease-gene, network, climate network, and DBLP datasets. Also, common neighbor similarity measures, Jaccard, Adamic/Adar, and PA measure were used.

In [9], Malik and his colleagues presented a method based on link prediction in social networks based on the structural and social information of the network. They used BlogCatalog, Flickr and Facebook datasets in their research. Also, common neighbor similarity measures, Jaccard, Adamic/Adar, and PA measure were used. Finally, they used AUC and accuracy metrics to evaluate the proposed method. Sharma and his colleagues in [17] presented a multilevel learning-based model for consumer preference in online social networks. They conducted their experiments on Amazon and Google+ datasets. Also, common neighbor and Jaccard similarity measures were used.

In [19], Yuan and his colleagues presented a prediction method based on graph kernel and SVM algorithm for social networks. In their research, they tested the simulation of the proposed method on the Epinions dataset and also used the accuracy evaluation criteria and F1-Score. They also used common neighbor similarity measures, Jaccard and Adamic/Adar. Yu and his colleagues in [18] presented a link prediction method to solve the cold start of users in multi-relational social networks based on network dependency analysis. Nassar et al. in [12] investigated a new multi-criteria collaborative filter model in recommender systems. In this paper, a new multi-criteria multiple filter model based on deep learning was proposed. In [11], Mashal et al. analyzed multi-criteria recommender systems in the Internet of Things. In this article, a recommender system was proposed using a hybrid multi-criteria decision-making approach based on the analytic hierarchy process and simple additive weighting methods. Berkani et al. in [3] investigated recommender systems in social networks and proposed a method based on the semantic and social display of user profiles.

In fact, most of the mentioned solutions have examined the problem in a scattered and static manner with emphasis on a specific application or strategy. In this research, a hybrid cumulative knowledge framework (HCKF) has been designed for friend recommendation in social networks. HCKF performs a huge amount of precise evaluations on data, including pre-processing on the data, data preparation, optimal feature selection and user clustering in the background and therefore, friend suggestions can be provided at an acceptable precision, accuracy and speed. HCKF solves the cold start challenge of new users, increases the accuracy of providing attractive friends to users and also minimizes the error rate. The experimental results demonstrate the advantages and effectiveness of the proposed framework in social networks. After simulating HCKF and calculating the error rate and accuracy, the final precision rate of the friend suggestion to the newly logged-in user is equal to 90%. Moreover, the average accuracy in the proposed framework is equal to 0.0197, which has improved significantly compared to other related methods. In the next section, we describe the proposed framework and related parts.

3 Proposed Framework

In this section, we describe the execution phases, workflows and functional capabilities of the HCKF, which is the hybrid recommender framework based on new knowledge and a cumulative learning system to introduce friends in social networks. The main goal of the HCKF is to suggest a friend in the social network to new users who are facing the cold start problem, using the demographic information of the users. New users are those users who have no function and activity in the system. In this framework, an attempt has been made to perform a huge amount of processing on the data, including pre-processing on the data, data preparation, optimal feature selection and primary clustering of users in the background. Therefore, friend suggestions can be provided at an acceptable performance, speed and accuracy. For example, the mentioned processes need to be executed only once in periods of one month or several months. On the other hand, in the category selection component, logging into the proposed framework is mandatory for each new user. Figure 1 presents the HCKF with related execution phases, the functional capabilities and workflows used in each part.

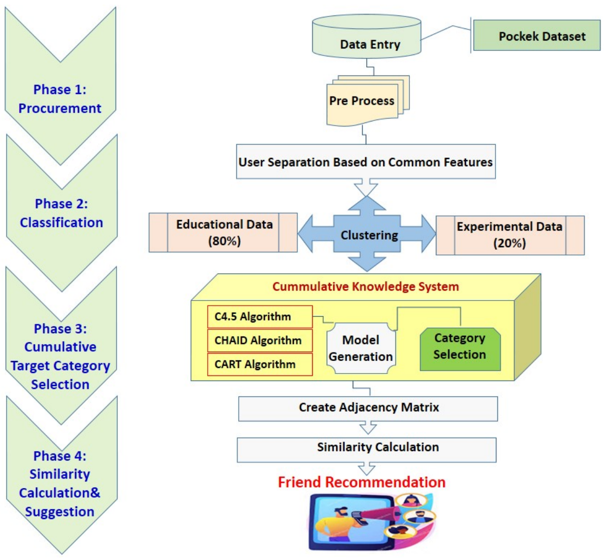

Figure 1: HCKF framework.

The operational phases and related workflows of the HCKF are as follows:

Phase 1: Procurement

In this phase, first, the intended dataset is entered into the framework by the Data Entry element and then pre-processing of the data is done on this information and data are converted into the acceptable format. Afterward, only the separated features are selected, and the rest of the features are removed. In this research, data related to Pokek users, along with 59 features, are entered into the proposed system. After entering the data, pre-processing of the data is done on the information. In this step, the strategy used is to extract users who have incomplete information. In HCKF, the incomplete features of each user are completed by the information provided by other similar users. With this strategy, users who have incomplete information will be completed and will not be problematic in the processing process. After applying pre-processing on the data, those users who have incomplete information and more than half of the features are completely removed from the data. In order to run the data in programming and data mining software, they must be converted into an acceptable format. In this phase, all data are mechanized in the form of text and Excel files. Finally, among all the features, only the separated features are selected, and the rest of the features are removed.

Phase 2: Classification

In this phase, the data from the previous step are entered into the X-Means clustering algorithm to apply primary clustering of the data. This algorithm is an extended version of the K-Means algorithm. In the X-Means algorithm, it is not necessary to enter K, and the algorithm itself selects the optimal number of clusters or K in the number of optimal iterations [22]. Therefore, at this stage, all users in the system are clustered based on their demographic information. This initial clustering is applied as a classification method on unsupervised data so that it can be used in cumulative learning algorithms for training models. After clustering the data, it is necessary to separate training and test data. For this purpose, 80% of the data will be separated as training data, and 20% of the data will be separated as test data. It should be noted that, up to this stage, it is done in the background of the system, and the user has no knowledge about it.

Phase 3: Cumulative Target Category Selection

After separation of the clustered data and in order to provide friend suggestions to new users, first all decision tree algorithms, including C4.5, CHAID, and CART, generate their models and create trees, based on the training data. In order to determine a category or a cluster for a new user, it is necessary to apply the cumulative learning system and enter the generated trees. This process is much more suitable than the case where the new user is determined by the clustering technique. Therefore, after determining the category related to the new user by each algorithm, the desired answer is entered into the core. Similar to the strategy used in the boosting system, in the collective learning system, the categories with the largest number are selected as the main category for the target user.

Phase 4: Similarity Calculation & Suggestion

In this phase, after selecting a category for the new user, all users in that category are placed in a proximity matrix with the new user. The rows of the matrix are the numbers of users, and the columns are their characteristics. Then, the similarity metric between users is calculated based on the similarity features, and finally, those users who have more similarity are suggested to the new user. In this step, the combination of similarity criteria is used. Therefore, more favorable and optimized offers can be made.

After describing the operational phases of the HCKF and related workflows in the next section, we intend to implement the details of the proposed framework and describe related algorithms.

4 Implementation and Evaluation

In this section, the simulation details for the HCKF are described and then the framework is evaluated, based on the some common key metrics. In the infrastructure layer, 64-Bit Operating System (Windows 10), 16 GB RAM and Intel Core™i7 CPU is used.

For simulation, the popular Slovak social network dataset was used. This dataset is known as Pokec, which consists of two general files. The first file presents the profile of 1632803 users.

http://snap.stanford.edu/data/soc-pokec.html

The second file also shows the friends and people with whom each user is connected [21]. There are more than 1.6 million records in this file. This file contains a collection of 59 features regarding personal information, work fields, interests, people characteristics and related values. In fact, the number of demographic characteristics of people is around 59. In general, this file contains two features that specify the origin and destination node or user. Since the data source has null and unused values and also includes some additional features, in the procurement phase, we removed the samples that included outlier data and also unused and additional features. Data pre-processing is done by Rapidminer data mining software and Excel software. The workflow is that we first identify the features that contain outliers through Rapidminer data mining software and then remove those features through Excel software or RapidMiner. Removing the useless features, they do not have a negative impact on the results. In this step, some fields that have different string values are analyzed and replaced with 0 and 1 values.

In the next step, the most optimal number of clusters is determined using Weka data mining software. The number of K=n samples (n is the optimal number of clusters) is applied to the K-Means algorithm, and finally, 2,000 users are clustered. It should be noted that there are different ways to cluster the data and select the inputs. In the current research, the selection of samples is 70% training data and 30% experimental data. At this stage, it is enough to select the K-Means algorithm from the clusterer and evaluate different K degrees by clicking on its name.

Finally, for all K value sets in the K-Means algorithm, we calculate the error rate and compare it to other values. Due to the weaknesses of the Weka software, this operation was performed by the Rapidminer data mining software. It was observed that the number of k=100 is the optimal number of clusters and, therefore, k=100 is considered as the default number of the clusters for the clustering algorithm.

In this part, using the CHAID decision tree algorithm, the previous output is divided into test data and training data. In order to train the decision tree, it is necessary to use a set of data as training data to generate the model. Also, a very small part of the data should be considered as test data. Therefore, we separate about 98% of the data as training data and about 2% as test data. 98% of the output of the clustering section is entered into the decision tree as training data. Necessary training rules are applied to the tree based on information_gain criteria. It is observed that only one of the categories is not among the main categories and is close to the similar category. Therefore, the precision of the selection method is around 90%. After implementing the decision tree model, a main decision tree is produced, which Figure 2. is only a small part of the main decision tree:

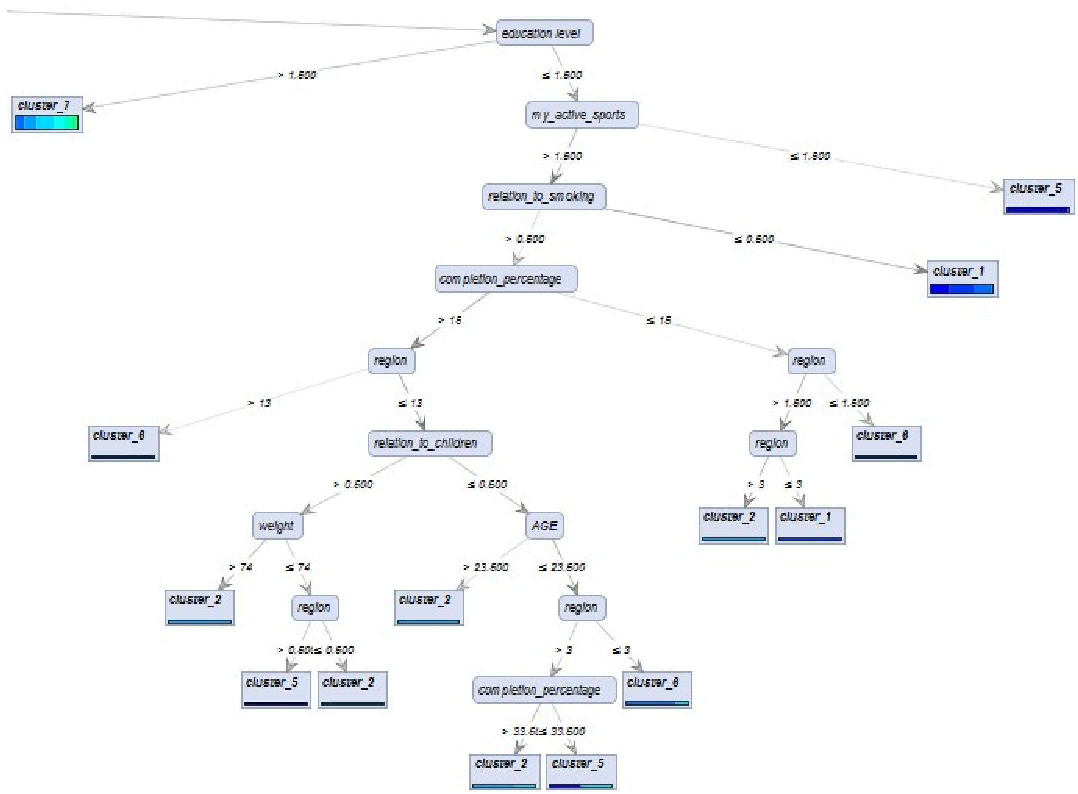

Figure 2: Generated decision tree schema.

For the similarity measurement, after selecting the desired cluster for the new user, similar neighbors with each user $n_j \in N$ should be extracted. Neighboring users are set in the form of an array named NG, and then, the similarity between the new users and similar users in the NG list should be calculated. The final prediction is based on the scores of neighboring users. The results of the similarity criteria are based on the demographic characteristics of the users, which are shown by set D. The final equation to calculate the degree of similarity between the new user and neighboring users is obtained by the following equation:

$$\mathrm{sim}(n,u) = \frac{\sum_{j=1}^{I} SF_j * w_j}{\sum_{j=1}^{I} w_j} \tag{4.1}$$

where SF_j is the similarity value of the jth feature and wj is the weight of the desired feature. To calculate the user similarity, the features of age, gender, occupation, etc. have been used, and each of them has been given a weight according to the importance. For example, a weight of 0.5 has been assigned to the age attribute, 0.25 to the gender attribute and 0.25 to the job attribute. The values of the above weights can be changed, but their sum must be 1. Therefore: $D = \{d_1 = age, d_2 = gender, d_3 = Occupation\}$ and the set of weights are: $W = \{w_1 = 0.5, w_2 = 0.25, w_3 = 0.25\} = 1$.

In this research, a hybrid similarity metric is used to calculate the degree of similarity between users. For each feature d_j we define a function $SF(at_1, at_2)$ that has values between $[0, 1]$. This function calculates the degree of similarity of two features related to a pair of users. According to the nature of the features, there are the following categories for features:

- Numerical features: For numerical features such as age, we have considered a similarity measure as follows:

$$w_{age} = \begin{cases} \left(1 - \frac{|\mathrm{Diff}|}{\mathrm{Diff}_{\max}}\right)^{\beta} & \mathrm{if}|\mathrm{Diff}| \leq \mathrm{Diff}_{\max} \\ 0 & \mathrm{if}|\mathrm{Diff}| > \mathrm{Diff}_{\max} \end{cases} \tag{4.2}$$

In the above equation, Diff presents the age difference between users, and $\mathrm{Diff}_{\max}$ is a maximum difference value defined by the researcher. If the researcher wants to increase the wage weight value compared to the Diff value, it is enough to set the β value less than 1.

- String features: Some features of the Slovak social network dataset are expressed in the form of string values. Therefore, the following formula is used to calculate the similarity of these features.

$$w_{str\_attr} = \frac{|n_j.StrAttribute \cap u_j.StrAttribute|}{MAX((LEN(n_j.StrAttribute), LEN(u_j.StrAttribute))} \tag{4.3}$$

N_j represents the new user and u_j represents the existing user in the cluster. As it is clear from the above formula, the string attribute of two users is received and placed in the string array format.

- Boolean features: For Boolean values, the following similarity criterion is used:

$$W_{\mathrm{gender}} = \begin{cases} 1 & \text{if } att1 == att2 \\ 0 & \text{if } att1 <> att2 \end{cases} \tag{4.4}$$

A value of 1 or 0 is returned depending on whether the values of att1 and att2 are the same. Finally, all the similarity metrics are combined, and a number between 0 and 1 is calculated, which indicates the degree of similarity between two users in the social network, using the proposed method. After adding the new user to the corresponding cluster through the C4.5 decision tree, all the hybrid similarity metrics are applied to the new user with all the existing users in the target cluster. After calculating the degree of similarity of the new user with other users and according to a threshold, a number of friends with the highest degree of similarity are suggested to the new user. Afterward, the new user is added to the graph in the social network and his friends are also determined, then the friendLink algorithm [13], which is one of the most popular link prediction algorithms, is applied to the new graph and combines other friends with the provided friends and it is presented as the final suggestions to the new user.

In the following, it should be noted that one of the most important techniques used for clustering large datasets and social networks is the K-means clustering algorithm, which was examined in the previous section. As discussed, one of the most important defects of this algorithm is the determination of the number of cluster Ks. In Figure 3, the amount of error is shown as a graph for each value of K.

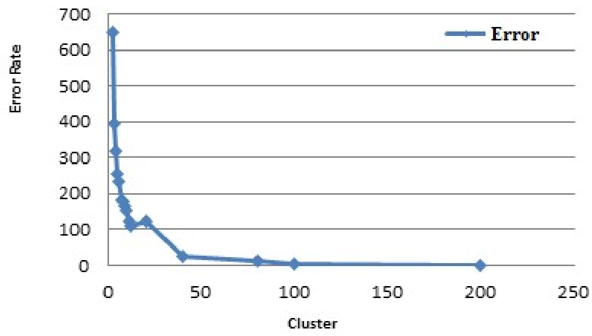

Figure 3: Comparison of clustering error for different number of Ks.

As can be seen from Figure 3, the amount of clustering error is specified for each K. Therefore, considering that the number of clusters K=100 has the least error, we will consider the number of clusters 100 for clustering and continuing the simulation and the obtained results. In Figure 4, the error rate of choosing the right cluster by decision tree algorithms is compared.

As can be seen from Figure 4, the amount of clustering error is specified for each K. Therefore, considering that the number of clusters K=100 has the least error, we will consider the number of clusters 100 for clustering and continuing the simulation and the obtained results. In Figure 5, the error rate of choosing the right cluster by decision tree algorithms is compared.

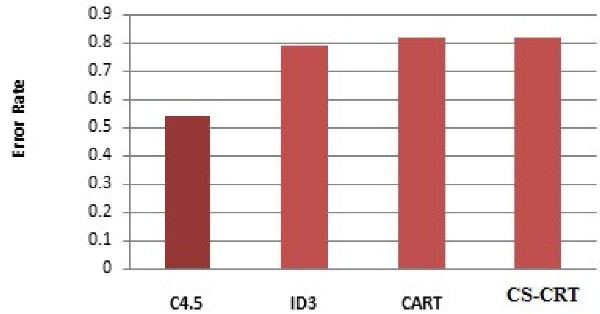

Figure 4: Error rates of choosing the right cluster using decision tree algorithms.

As it can be seen, the error rate of the C4.5 decision tree algorithm is less than other methods, and the accuracy of this method is much higher. Therefore, the C4.5 algorithm can be used as the basic decision tree algorithm for the current research to select the appropriate cluster. In the next section, the simulation results of the proposed method and comparison with related work are discussed.

5 Discussion and comparison

In this section, to present the efficiency of the proposed framework, some key and common performance metrics such as error rate, precision and accuracy have been compared in the HCKF with related work. First, the error rate of various decision tree algorithms is compared in Table 1. The accuracy of the decision tree algorithms is also shown in Figure 5.

Table 1: Comparing the error rate of choosing the right cluster using decision tree algorithms

C4.5 Algorithm	ID3 Algorithm	CART Algorithm	CS-CRT Algorithm
0.5411	0.793	0.818	0.818

As can be seen, the error rate of the C4.5 decision tree algorithm is less than other methods, and the accuracy of this method is much higher than all other methods. Therefore, the C4.5 algorithm can be used as the basic decision tree algorithm for the current research to select the appropriate cluster. After simulating the proposed approach and

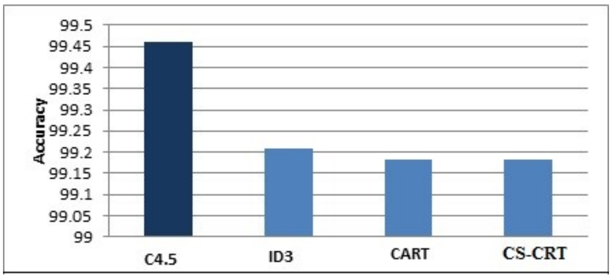

Figure 5: Comparing the accuracy of choosing the right cluster using decision tree algorithms.

calculating the error rate and accuracy, the final precision rate of the friend suggestion to the newly logged-in user is equal to 90%. To calculate the accuracy, a user in the system whose number of friends is already known was selected, and after implementing the proposed method, the obtained results were compared with real friends. After comparing the results, it was observed that 90% of the friends are exactly the same and 10% of the suggested friends are not among the new user's previous friends. Figure 6 presents the accuracy and error rate of the proposed approach.

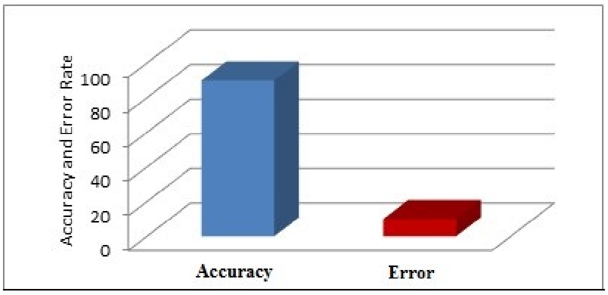

Figure 6: Accuracy and error rate of the proposed approach.

Moreover, in Figure 7 that extracted from the simulation, the accuracy, error rate and MAP metric are shown.

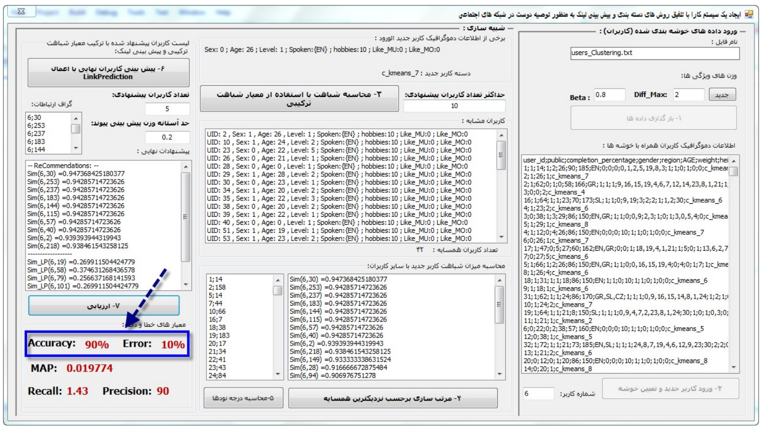

Figure 7: Results of accuracy, error rate and MAP metric of the proposed approach.

As it can be seen from Figure 7, the most important validation metrics for such recommender systems are Precision and Recall. The Precision metric can be calculated using the following equation:

$$\text{precision} = \frac{TP}{TP + FP}. \tag{5.1}$$

According to the above relation, TP is the number of recommendations that are correctly recognized. FP is also the number of proposals that are not correctly recognized. As can be seen, the precision rate of the proposed method is 90%.

Therefore, each new user in social networks receives a set of users as suggested users according to the used dating algorithm. This set of users, whose number is k samples, is known as top-k. After a set of top-k users is proposed to a new user or a test user, the MAP measurement metric is used to calculate the accuracy of the proposed approach to suggest friends to newly logged-in users. In the following equation calculation of the average accuracy of the proposed approach is shown [16].

$$MAP = \frac{1}{|N|}\sum_{u=1}^{|N|}\frac{1}{r_n}\sum_{k=1}^{r_n}\text{precision}(u,k). \tag{5.2}$$

According to the above formula, N represents the number of users in the dataset, and r_n represents the number of users corresponding to the new user n. Precision (u, k) also expresses the precision in the kth position or location of the user list suggested for user n. It should be noted that the MAP criterion is a combination of recall and precision. In Figure 8, the average accuracy comparison of the most popular algorithms for link prediction is shown.

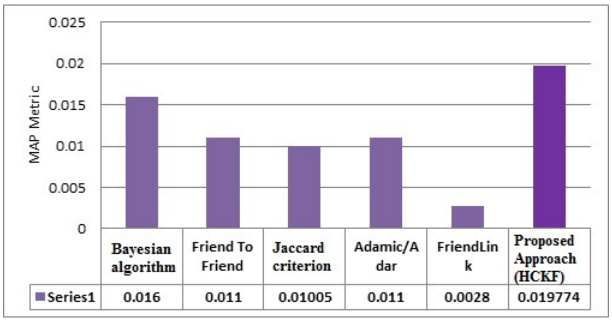

Figure 8: Comparison of the average accuracy of the proposed method with related work.

It can be seen from Figure 8, the average accuracy (MAP) of the friend's suggestion in the proposed method is equal to 0.019774, which has been significantly improved compared to other related methods, which have been used recently on the same dataset. On average, the improvement of friend suggestion accuracy in the HCKF compared to FriendLink, Adamic, Jacquard, FOF, and Bayesian methods is about 7.06%, 0.17%, 196%, 1.79% and 1.23%, respectively. Moreover, a report of the results obtained from different implementations of the proposed approach is shown in Table 2, according to the increase in the number of users.

Table 2: A summary of the final results based on the number of users

Number of Users	Accuracy	Error Rate	Precision	Recall	MAP
100	75.02	24.98	50	22.4	0.02
200	74.24	25.76	55	16	0.01
300	75.8	25.76	65.6	11.66	0.006666
400	76.95	23.05	58.9	10.75	0.005
500	76.87	23.13	65	11.2	0.004
600	78	22	67.5	11.33	0.0033
700	78.95	21.05	70	11.1	0.003184

HCKF performs a huge amount of precise evaluations on data, including pre-processing on the data, data preparation, optimal feature selection and user clustering in the background and therefore, friend suggestions can be provided at an acceptable precision, accuracy and speed. HCKF solves the cold start challenge of new users, increases the accuracy of providing attractive friends to users and also minimizes the error rate. The experimental results demonstrate the advantages and effectiveness of the proposed framework in social networks. After simulating HCKF and calculating the error rate and accuracy, the final precision rate of the friend suggestion to the newly logged-in user is equal to 90%. Moreover, the average accuracy in the proposed framework is equal to 0.0197, which has improved significantly compared to other related methods. Therefore, according to the simulations performed on the proposed approach and related algorithms in the same conditions, it is observed that the proposed framework suggests friends in the social network with higher efficiency and accuracy.

6 Conclusion

The recommendation system is an effective technique to solve the information overload problem and is one of the most common applications of today's big data and communication technologies. Friend recommender is a popular service that has been widely developed and focuses mainly on relationships and user interests. Traditional friend recommender systems suffer from some shortcomings that hamper their effectiveness. Hybrid techniques to recommender systems can significantly improve the overall quality of proposals. In this research, a hybrid recommender system has been designed based on new knowledge patterns and cumulative learning to recommend friends in social networks. To present the efficiency of the proposed framework, some key and common performance metrics such as error rate, precision, accuracy and MAP have been compared in the HCKF with related work. Based on the simulation results, the final accuracy rate of the friend suggestion to the newly logged-in user is equal to 90%. The average precision of friends' suggestions in the proposed approach is equal to 0.0197, which is a significant improvement compared to other methods that have been recently used on the same dataset. On average, the improvement of friend suggestion accuracy in the proposed method compared to Friend Link, Adamic, Jacquard, FTF and Bayesian methods is about 7.06%, 0.17%, 196%, 1.79% and 1.23%, respectively. Therefore, by simulating the proposed method on the Slovak social network, it can be concluded that by using the HCKF, a significant improvement can be achieved in providing friend suggestions to new social network users. The use of other methods and algorithms for intelligent recommendation and other effective similarity metrics can be the future work of this research.

References

[1] M. Bazargani and S.H. Alizadeh, *A deep neural network-based approach in tag recommender system to overcome users' Cold Start*, Int. J. Nonlinear Anal. Appl. **15** (2024), no. 7, 197–214.

[2] M. Bazargani and E. Namazi, *A study on a combined model in business intelligence for improving electronic insurance*, Int. J. Bus. Intell. Res. **6** (2015), no. 1, 49–55.

[3] L. Berkani, S. Belkacem, M. Ouafi, and A. Guessoum, *Recommendation of users in social networks: A semantic and social based classification approach*, Expert Syst. **38**, no. 2 (2021), e12634.

[4] L. Chen, M. Gao, B. Li, W. Liu, and B. Chen, *Detect potential relations by link prediction in multi-relational social networks*, Decision Support Syst. **115** (2018), 78–91.

[5] Y. Deldjoo, T. D. Noia, and F. A. Merra, *A survey on adversarial recommender systems: From attack/defense strategies to generative adversarial networks*, ACM Comput. Surv. **54** (2021), no. 2, 1–38.

[6] E. Kozegar, H. Yarmohammadi, M. Bazargani, and Z. Homayounpour, *Presenting a novel method based on collaborative filtering for nearest neighbor detection in recommender systems*, Intell. Multimed. Process. Commun. Syst. **1** (2020), no. 1, 55–64.

[7] N. Li, Q. Huang, X. Ge, M. He, S. Cui, P. Huang, S. Li, and S.F. Fung, *A review of the research progress of social network structure*, Complexity **2021** (2021), no. 1, 6692210.

[8] Z. Li, F. Xiong, X. Wang, H. Chen, and X. Xiong, *Topological influence-aware recommendation on social networks*, Complexity **2019** (2019), no. 1, 6325654.

[9] S. Mallek, I. Boukhris, Z. Elouedi, and E. Lefèvre, *Evidential link prediction in social networks based on structural and social information*, J. Comput. Sci. **30** (2019), 98–107.

[10] K. Madadipouya and S. Chelliah, *A literature review on recommender systems algorithms, techniques and evaluations*, Broad Res. Artific. Intell. Neurosci. **8** (2017), no. 2, 109–124.

[11] I. Mashal, O. Alsaryrah, T.Y. Chung, and F.C. Yuan, *A multi-criteria analysis for an internet of things application recommendation system*, Technol. Soc. **60** (2020), 101216.

[12] N. Nassar, A. Jafar, and Y. Rahhal, *A novel deep multi-criteria collaborative filtering model for recommendation system*, Knowledge-Based Syst. **187** (2020), 104811.

[13] A. Papadimitriou, P. Symeonidis, and Y. Manolopoulos, *Friendlink: link prediction in social networks via bounded local path traversal*, Int. Conf. Comput. Aspects Soc. Networks, 2011, 66–71.

[14] V.S. Parvathy and T. K. Ratheesh, *Friend recommendation system for online social networks: A survey*, Int. Conf. Electronics Commun. Aerospace Technol., Vol. 2, 2017, pp. 359–365.

[15] J. Piao, G. Zhang, F. Xu, Z. Chen, Y. Zheng, C. Gao, and Y. Li, *Bringing friends into the loop of recommender systems: An exploratory study*, Proc. ACM Human-Comput. Interact., Vol. 5, 2021, pp. 1–26.

[16] S.H. Shalforoushan and M. Jalali, *Link prediction in social networks using Bayesian networks*, Int. Symp. Artific. Intell. Signal Process., 2015, pp. 246–250.

[17] P.K. Sharma, S. Rathore, and J.H. Park, *Multilevel learning based modeling for link prediction and users' consumption preference in Online Social Networks*, Future Gen. Comput. Syst. **93** (2019), 952–961.

[18] S.Y. Wu, Q. Zhang, C.Y. Xue, and X.Y. Liao, *Cold-start link prediction in multi-relational networks based on network dependence analysis*, Phys. A: Statist. Mech. Appl. **515** (2019), 558–565.

[19] W. Yuan, K. He, D. Guan, L. Zhou, and C. Li, *Graph kernel based link prediction for signed social networks*, Inf. Fusion **46** (2019), 1-10.

[20] H. Yuan and A.A. Hernandez, *User cold start problem in recommendation systems: A systematic review*, IEEE Access **11** (2023), 136958–136977.

[21] J. Yuan, W. Wu, Y. Li, and D. Du, *Active friending in online social networks*, Proc. Fourth IEEE/ACM Int. Conf. Big Data Comput. Appl. Technol., 2017, 139–148.

[22] H. Zarzour, F. Maazouzi, M. Al-Zinati, A. Nusayr, M. Alsmirat, M. Al-Ayyoub, and Y. Jararweh, *Using K-means clustering ensemble to improve the performance in recommender systems*, Int. Conf. Intell. Data Sci. Technol. Appl., 2022, pp. 176–180.

[23] Z. Zhang, J. Wen, L. Sun, Q. Deng, S. Su, and P. Yao, *Efficient incremental dynamic link prediction algorithms in social network*, Knowledge-Based Syst. **132** (2017), 226–235.

[24] H. Zou, Z. Gong, N. Zhang, and W. Zhao, *TrustRank: A cold-start tolerant recommender system*, Enterprise Inf. Syst. **9** (2015), no. 2, 117–138.