AN ENHANCED ITEM RECOMMENDATION APPROACH USING THE SIGMOID FUNCTION AND JACCARD SIMILARITY COEFFICIENT

[1,*]Shiraz Ismail, [2] Alhassan Abdul-Barik & [3]Salifu Abdul-Mumin

[1]Computer Science Department,
Tamale Technical University,
Tamale, Ghana

[2]Computer Science Department
University for Development Studies,
Tamale, Ghana

[3]Department of Information Systems & Technology,
C. K. Tedam University of Technology and Applied Sciences,
Navrongo, Ghana

[*]Corresponding Author Email: sismail@tatu.edu.gh

Abstract

Based on prior interactions between users and service items, recommendation systems have developed into practical tools for filtering and obtaining vital data. These systems are often used in a range of commercial industries, including e-commerce, tourism, social networking, and academic research. Collaborative filtering, content-based filtering, and hybrid recommender systems are the three main categories of recommender systems. Collaborative filtering recommender systems, which presume that users would be interested in products that users similar to them have highly rated, take into consideration users' tastes (in terms of item preferences). Content-based filtering recommender systems base their recommendations on the textual content of a product, using the assumption that customers would prefer items that are similar to those they have previously enjoyed. In a hybrid recommender system, two techniques are combined. These systems struggle with scalability, data sparsity, and cold starts, which leads to low-accuracy prediction and coverage. In this study, we proposed a unique recommendation method and applied the sigmoid function to the Jaccard similarity index. In our proposed method, which included the rating significance of items, we used the sigmoid function on the Jaccard similarity index to evaluate the asymmetry relationship between users. The similarity between the target user and his or her neighbours is then assessed using the asymmetric association and rating significance. Several cutting-edge similarity metrics are evaluated using experiments on three real-world data sets. The results show that the new similarity model performs better than the baseline models in terms of diversity and prediction accuracy.

Keywords: *Recommender System, Accuracy, Diversity, Sigmoid function, Jaccard Similarity coefficient*

INTRODUCTION

A recommender system (RS), also known as a recommendation engine, is a type of information filtering system that identifies items most relevant to a certain user.The growth and expansion of research have had a substantial influence on user and researcher habits, allowing people to learn more about research articles and items. It may be challenging for researchers and users to choose the article or item that best matches their needs. Recommender systems are utilised in a variety of fields, including e-commerce (Alamdari et al., 2020), health (Galeano & Paccanaro, 2018), social networks, industry (Zhou et al., 2007), electronic learning (Tarus et al., 2018), music (Katarya & Verma, 2018), Internet of Things (IoT), food and nutrition information system, and marketing (Jothilakshmi & Thangaraj, 2018). To establish automation of item suggestion in an e-commerce environment, they employ both traditional and modern methods of recommender systems (Li et al., 2019), as a machine learning technique (Shoja & Tabrizi, 2019).

Based on their prior experiences, recommender systems frequently attempt to predict and recognise users' preferences for certain items. Figure 1 shows the high-level design of a typical RS. When a user interacts with a system, he or she gives explicit or implicit input about his or her preferences (e.g., likes, clicks, ratings).

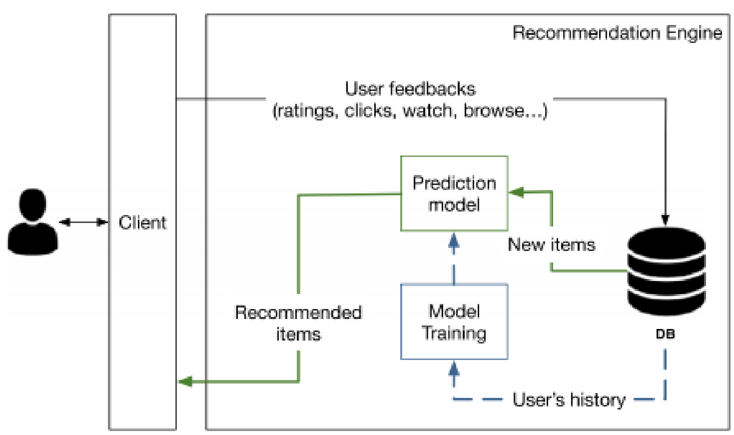

Figure 1: Recommender System architecture (Campana & Delmastro, 2017)

Sigmoid Function

A Sigmoid function is a mathematical function that can map any real value to a number between 0 and 1, and is shaped like the letter "S". The logistic sigmoid function stated in Equation 1 is one of the most prevalent sigmoid functions.

$$S\,(x) = \frac{1}{1+e^{-x}} \tag{1}$$

The use of sigmoid functions is advantageous for many machine learning applications that need the transformation of a real number to a probability. By using a sigmoid function as the last layer, a probability score that is easier to use and understand may be produced from the output of a machine learning model.

Jaccard Similarity Index

The Jaccard similarity coefficient, commonly referred to as the Jaccard similarity index, analyses entities from two groups to identify similarities and differences. The percentage scale, which measures how comparable the two sets of data are, runs from 0% to 100%. The closer the two groups are, the higher the percentage. The Jaccard similarity is calculated by dividing the size of the intersection by the size of the union of two sets, as shown in Equation 2.

$$J\,(A,B) = \frac{|A \cap B|}{|A \cup B|} \tag{2}$$

Where $J = Jarccard\ distance$

$A = Set\ 1$

$B = Set\ 2$

REVIEW OF RELATED WORKS

Myriad literary works have been produced to fix the many design problems that arise while creating systems that provide recommendations. The work of (Alamdari et al., 2020) generated an individualised recommendation and examined a mobile e-commerce Collaborative filtering (CF) based recommender system. Three significant systemic methods were examined by the researchers. Modules with input, suggestion, and output capabilities are provided. To estimate customer ratings for unclassified products, they developed the CF technique based on item classification forecasts. The researchers first evaluated item similarity before choosing the closest neighbours using a novel procedural similarity measurement. They also examined the CF based totally on object rating prediction to recover from the problem of consumer data scarcity in the mobile electronics trade. They also mentioned several issues that need to be resolved to protect the integrity and development of subsequent investigations.

To deal with new users' cold starts and the sparsity of user item ratings, (Saini et al., 2017) presents a random walk model for a recommendation that combines trust-based and collaborative filtering techniques. The random walk model (It considers not just the ratings of the target item but also the ratings of comparable items) allows a better assessment of confidence in a recommendation. The Epinions dataset was used for evaluation, and their model was compared to existing trust-based and collaborative filtering approaches. Because mobile phone-based social networks necessitate a distributed recommender, this rating is supposed to be kept in a centralised repository.

Additionally, by exploring transitive linkages based on user input data using association retrieval technology, (Chen et al., 2011) suggested direct and indirect user similarity, calculated the similarity matrix utilising the relative distance between the user's ratings, and improved the superiority of recommender systems to realise a novel collaborative filtering strategy to handle the sparsity problem. Finally, the author conducted an experiment utilising MovieLens data and established the efficacy of the proposed approach in curtailing the sparsity problem while maintaining a high coverage rate and recommendation quality. In this scenario, their method caused a data overload challenge and hindered the scalability of collaborative filtering recommender systems.

Big Data is proposed to be integrated into a CF-based recommendation method (Sun et al., 2014) based on MapReduce, it uses a suitable distributed computing strategy. The researchers employed cosine similarity and Pearson correlation in the article CF algorithm. This conclusion is crucial since shoppers seldom evaluate products in actual stores. The authors next looked at the use of RS for marketing, including choosing appropriate product pairings for stores and creating product advertising. In addition to the system's effectiveness for retail items and enterprises, it was found that it can manage enormous volumes of data, increase scalability, and provide a new source of market support for accuracy. This method requires the use of big data technologies, which makes it expensive and computationally challenging.

The complications that CF methods confront, including sparsity, scalability, and synonymy, are listed in (Rahnama et al., 2014). The project also shows how much RS has been customised. The CF approach uses user ratings, reviews, and data from the whole applicator to provide suggestions by comparing the active user's characteristics and preferences to those of the current applicator. Additionally, the CF method utilises similarity measurements like the Jaccard similarity and the cosine matrix. The author concluded that sales on significant e-commerce sites were declining despite a lack of statistics. He thus urged developers of recommender systems to forgo using neighbourhood algorithms when implementing recommender systems for huge websites. because it reduces metric coverage and has less recommendation accuracy. It is frequently used despite its drawbacks, which include restricted scalability, data sparsity, and the synonymy problem.

Pearson's Correlation, Cosine, and Adjusted Cosine are widely employed as CF technique measurements (Aprilianti et al., 2017). In (Patra et al., 2015) a novel similarity measuring methodology for certain criteria in CF is presented. For CF in dealing with sparse data, a novel similarity measure based on the Bhattacharyya coefficient was proposed.

Also, researchers in (Wenjun & Dong, 2015) provide a list-wise diffusion-based recommender method that weights the connections in a user-object bipartite network per the degree of similarity between users. In ListDB, the authors consider users' preference orders as a full instance, as indicated by observed temporal information, and quantify user-user similarity using the Jensen-Shannon divergence between users' probability distributions across permutations of regularly collected items. In the similarity computation, the top-k (k 1) probability model was also utilised to locate the nearest neighbour. Experiment findings on two benchmark datasets demonstrate that their proposed model may significantly increase the novelty and variety to provide more customised recommendations while maintaining high accuracy when K is large but reduced accuracy when K is low.

Furthermore, (Sun & Xu, 2016) developed and put into operation a specially designed recommendation system for use in e-commerce using a web mining method. Because online data is so complex, it might be challenging to employ data mining or other techniques to provide it as input for e-commerce recommender systems**.** The system assessed each e-commerce project's fundamental components, such as managing users, products, and orders, to give the best recommendations. However, the performance of the proposed method depends on computer resources like CPU power. As a result, performance optimization and development will be needed for more investigation into the viability of the RS implementation technique.

Using data samples from case studies on e-commerce, (Aditya et al., 2017) examine the effectiveness of memory-based and model-based CF approaches The accuracy, computation speed, and recommendation relevance of model-based RS exceed those of memory-based RS, according to the authors' evaluation of each technique using memory and user-based testing. The investigation did not take into account other RS performance factors including variety, serendipity, coverage, and newness.

For e-commerce recommender systems, (Aprilianti et al., 2017) developed and implemented a weighted parallel hybrid strategy integrating CF and CBF techniques. Additionally, the author claimed that the strategy may address the drawbacks of these two approaches, including CF's cold-start issue and CBF's lack of a variety of products to offer clients (Mansoury & Shajari, 2016). However, the method has a slow response time and a high operating expense.

To improve RS's efficiency, (Saini et al., 2017) developed a technique to recognise the events that follow consumer transactions. The study suggests using sequence pattern mining since some items are frequently acquired in phases. The researcher solely examines the customer profile and the item description, omitting consumers who haven't completed any purchases, which is the concept's drawback. With this approach, recommendations are more effective and responses are sent quickly. On the other hand, customers who had not made any purchases were not considered.

Researchers (Son & Kim, 2017) point to a CBF algorithm built on a multi-attribute network analysis that considers similarities between items that are not connected physically. The proposed technique has been tested with "MovieLens," and it resolves data sparsity and over specialisation problems while displaying robustness. Additionally, because the recommended solution only uses rating data obtained from the user's foreknowledge, it is unaffected by the cold start issue.

Also, a new technique (Wang et al., 2017) uses the Kullback-Leibler (KL) divergence approach in the signal process and information theory domain to calculate the similarity between items in their suggested model. A non-linear function is employed to better fit the interaction between users, which successfully compensates for the linear formula's weakness. To distinguish between users, they include two weighting components in their technique. The first element, the preference factor, considers global information on user preference behaviour. The second element, the asymmetric factor, is utilised to underline the users' unequal connection. A confirmation of the efficacy of the proposed user similarity model is verified via various tests on various datasets and proved to be appropriate for sparse data and increasing the prediction accuracy and quality of recommendation. The test results revealed that their superior similarity measure outperformed the commonly used similarity measures. However, because they employed all possible permutations of rating pairs in the computation of their similarity measure, this strategy has a higher computational complexity.

Authors (Hwangbo et al., 2018) expanded on the already-existing CF techniques by proposing procedures to educate fashion accessories and consumers to use fashion items' qualities while generating suggestions for fashion retail e-commerce. Their study examined ways to buy these things both offline and online. The writers concluded that a consumer's intended purchase will take the place of the product they have previously decided on. To show off the effectiveness of the system, they installed it in a real shopping centre. In terms of efficiency, the proposed method outperforms the collaborative filtering system.

A slope one technique based on a variety of user characteristics was developed by (Jiang et al., 2018). Their method consists of three steps: gathering reliable data, determining user similarity, and utilising the steep slope of the similarity and weight component of one algorithm. They also utilised the Amazon dataset to show that the new method, while taking more time to compute than the old slope one methodology, is more accurate.

To determine the average of the rated songs, root mean square error is used, (Mohamed et al., 2019) to solve the problem of cold start during recommendations. Researchers used the world's largest online music data sets for their research, which included explicit rating records for items identified by hybrid feedback as well as implicit and explicit user interaction records. To track each purchase made for each transaction, the authors created an association rule. and then used the cosine vector approach to compute similarities and produce recommendations. Evaluations of accuracy, recall, and F-metrics are employed when the data is limited to encourage users to try something new. The technique outperforms collaborative filtering algorithms in several experiments in terms of performance and accuracy measures. The recommendation engine, however, was unable to handle highly rated users and items as well as diverse dataset kinds like movies.

Considering the similarity between the category's characteristics and the object's visual, (Cao et al., 2019) uses a cooperative filter recommendation technique based on the matrix factorisation model. First, the authors predicted and filled in the missing evaluation items using a matrix factorization model based on user preferences. Using the neural network VGG16, the article's visual characteristics and category attributes were retrieved. The article's neighbours were then identified after being merged to establish how comparable the new article and the historical components are. Then, based on the anticipated new subject's score, which is based on the similarity between the new topic and its neighbours, they advise the user to choose the N subjects with the highest score. Image blending lessens the cold start of new elements by incorporating collaborative filtering and a matrix factorisation technique. The method offers constructive advice for systems with plenty of text and graphics, but it is unable to tackle the issue of new articles' cold starts.

To solve the issue of a cold start for new users, (Yadav et al., 2020) designed a hybrid recommendation system. The concept integrated elements of social networks, teamwork, and an improved user profile made from connected open data. It is also possible to anticipate ratings for items that have not yet received ratings by calculating item similarity. The researchers used ontology and collaborative features to address the problems with accuracy and computation time in recommendation systems. The authors' research, on the other hand, did not examine how user preferences might impact novelty, variety, or other criteria.

MATHEMATICAL FORMULATION OF THE PROBLEM

The ability to filter and deliver appropriate information to a user has become a possible challenge as a result of information overload. This dilemma emphasizes the necessity for information extraction algorithms that can filter unknown data and forecast whether a user would appreciate a particular resource. Recommender systems are a type of system that does just that.

In our approach, we have a set of ser $U = \{u_{1,\dots\dots\dots} u_N\}$ and a set of item $I = \{i_{1,\dots\dots\dots} i_M\}$. Each user u rates a set of items $RI_u = \{i_{u1,\dots\dots\dots} i_{uk}\}$. The rating of the user on item i is denoted $r_{u,i.}$ $r_{u,j}$ can be any real number, but often ratings are integers in the range [1,5], where 1 and 5 mean the lowest and highest rating respectively.

METHOD OF SOLUTION

The proposed algorithm is presented in this section; which addresses the accuracy and diversity concerns of service recommendations. The model consists of two main components: the asymmetry relationship and rating significance components.

Table 1: A Sample User/Item rating matrix

	Item				
User	**I_1**	**I_2**	**I_3**	**I_4**	**I_5**
A	-	5	-	1	-
B	4	3	5	2	4
C	-	5	1	1	4
D	4	3	-	-	-

Considering Table 1, it can be seen that user B is similar to user D since they both rated similar items (i.e., items I_1 and I_2). Considering only the two co-rated items without the other item ratings, we can say that the similarity between users B and D is symmetric (i.e., Sim (B, D) = Sim (D, B)). However, further research in recommender systems shows that combining ratings from the co-rated items and other unrated items can lead to improvement in item recommendation (Liu et al., 2014). Looking at the rating vector of B (i.e. (4, 3, 5, 2, 4)) and that of D (i.e., (4, 3, -, -, -)) in Table 1 shows that all ratings in D can be found in B, but the reverse is false. This means *B* has a significant impact on D and D has a limited influence on B. making the similarity asymmetric. Therefore, it is better to consider user similarity in asymmetry mode rather than in the usual symmetry mode.

The researcher uses the sigmoid function on the Jaccard similarity index to estimate the asymmetry relationship between the user A and B in Table 1. This is shown in Equation 3;

$$As\,(A,B) = \frac{1}{1+\exp\left(\frac{-|IA \cap IB|}{|IA \cup IB|}\right)} \tag{3}$$

To enhance the impact of the co-rated items on user *A*, than on user *B*, I_A the percentage of user A's co-rated items is increased above that of user *B*. I_B. The asymmetry of user similarity is defined by the difference between user *A* and user *B*. Assume $I_A < I_B$, then $As(A,B)$ demonstrates that the influence of user *A* on user *B* is weaker than the influence of user *B* on user *A*. As a result, $As(A,B)$ calculates a user's asymmetry similarity. Furthermore, taking into account the significance of a rating might have a positive impact on item recommendations (Wang et al., 2017). For example, consider the ratings of users A and C in Table 1, the ratings of (A, C) on item I_2 are (5, 5) and that on item I_4 is (1, 1). According to (Yang et al., 2014) this form of rating is more significant than their ratings on items I_3 and I_5 which are (-, 1) and (-, 4) respectively. In the user item graph, there should be a method to successfully fit the non-linear user relationships in order to accomplish this. To begin, we use Equation 4 to determine the rating significance (Sig). This is done in order to calculate how much the rating pair will impact the final rating.

$$Sig(i,j) = \frac{1}{1+\exp(-|R_{Ai}-R_m| . |R_{Aj}-R_m|)} \tag{4}$$

where R_m is the mean rating, R_{Ai} is the rating user *A* gives to item *i*, and R_{Aj} is the rating score of item *j* given by user *A*. As the difference between the ratings in the pair and the mean rating widens, the rating pair becomes more significant.

The researcher combined the rating significance and the asymmetry similarity between users as shown in Figure 2 by applying the sigmoid function to form our proposed model to improve the prediction accuracy.

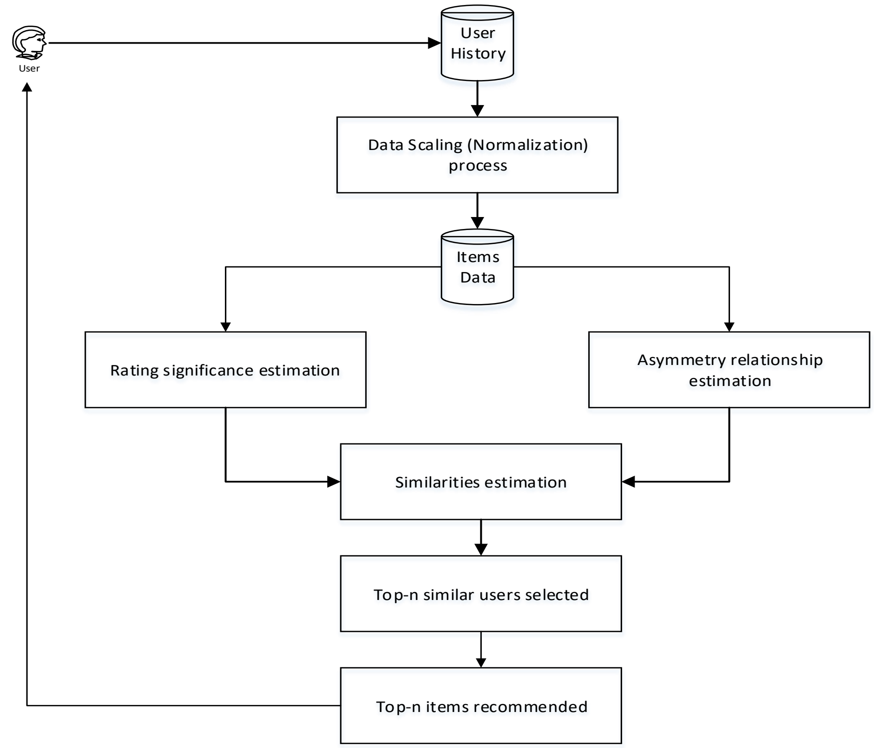

Figure 2 Flowchart of the proposed recommendation method.

The final formula for user similarity is presented using Equation 5.

$$S(A,B) = \frac{1}{1+\exp\left(\frac{-|IA \cap IB|}{|IA \cup IB|}\right)} * Sig(A,B) \tag{5}$$

3.1 Estimating the prediction values

We must first get user K -nearest neighbours to determine the forecast score a user A will offer an unrated item. The similarity values between user A and other users may be calculated using Equation (5). These similarity values may then be used to build a neighbour set based on user $A's$ first K closest neighbours. Equation 6 is used to compute the predicted value.

$$P_{(A,i)} = \bar{r}_A + \frac{\sum_{B\in K}(S(A,B)(r_{Bi}-\bar{r}_B)}{\sum_{B\in K}|S(A,B)|} \tag{6}$$

where $\bar{r}_A$ is the mean rating of user A.

DATASETS

Our experiments were carried out on three real-world datasets, namely, Epinions, Netflix, and MovieLens–100K, as shown in Table 3. Epinion (Massa & Avesani, 2006) is a site that allows visitors to review items and also determines whether to buy a particular item or not. This network also allows users to form social networks based on their interest areas or items reviewed. The Epinions dataset contains 7,649 items and 4,066 users. Netflix (Chen et al., 2017) is a company that rents DVDs. This company made available the Netflix dataset for its prize challenges on its site (Netflix.com). We extract a smaller sample of the dataset. This sample contains links with ratings greater than or equal to 3, using a 1 to 5 rating scale. The Netflix dataset sample is made of 5,640 items, 10,000 users, and 701,947 links with a sparsity of 0.0124. MovieLens (http://www.movie-lens.umn.edu) is a recommendation site established by the GroupLens Research Project at the University of Minnesota. The MovieLens-100K dataset contains 1,682 items, 943 users, and 100,000 links with a sparsity of 0.063.

Table 2 Details of the real-world rating data sets used

Dataset	Number of Users	Number of Objects	Number of Links	Sparsity
Epinions	4,066	7,649	154,122	0.00496
Netflix	10,000	5,640	701,947	0.0124
MovieLens-100K	943	1,682	100,000	0.063

Experimental Settings

To examine the performance of the proposed algorithm, commonly established settings used in item recommendation experiments are adopted for evaluation. We randomly split the user-item edges in each dataset into two; a train set and a test set. The train set contains 90% of the dataset, and the test set 10%. Several experiments are conducted using the four methods (i.e., ListDB, MBD, TrustWalker, and the proposed method). Each experiment is 100 times repeated independently on the train, and test sets and the average results are obtained for each metric.

Evaluation Metrics

In our proposed recommendation algorithm, the top-K items are recommended to the target user from a well-ordered list of unrated items. To evaluate the efficiency of recommendation, one diversity metric, one novelty metric, and five accuracy metrics are used to evaluate the model performance.

The Area Under the receiver operating characteristic curve (AUC)

This metric is used to evaluate the accuracy of a recommender model (Zhou et al., 2009) AUC represents the probability that an unrated randomly picked item characterised a lower rank than a rated randomly picked item in item recommendation. The resources acquired by both unrated and rated item pairs after *m* independent comparisons are used to calculate the AUC value based on Equation 7; The greater the AUC, the better the diagnostic test's overall performance

$$AUC = \frac{1}{n}\sum_{u_x=1}^{n}\frac{m_1 + 0.5m_2}{m} \tag{7}$$

where m represents the total number of independent comparisons, m_1 represents the total number of times that the rated items have resources higher than the unrated ones, and m_2 indicates the total number of times the unrated items have resources totalling the same as the resources rated by the target user u_x. A higher AUC score represents a greater accuracy of recommendation. If scores from independent distributions are generated, then the score measured by AUC will be around 0.5, making it analogous to a random choice. Therefore, a greater AUC score beyond 0.5 is a suggestion of how superior the technique is as compared to a random choice.

Ranking Score (RS)

A ranking score is used to measure the capacity of recommendation algorithms to rank items and make recommendations to a target user based on the preference of the user (Zhou et al., 2007). For each user-item edge in the data set, the ranks of all the items in the recommendation list of a user are generated. The test data set's user-item relationships are averaged to get an average RS score, which is then used to estimate how accurate the recommender system will be. An algorithm's performance and accuracy are better and more accurate when its ranking score is lower. Mathematically, for a target user u_x, the RS is given as in Equation 8;

$$RS = \frac{1}{|E^P|} \sum_{(u_x, s_\alpha) \in E^P} \frac{p_{s\alpha}}{k_{ux}} \tag{8}$$

where the size of the test set is represented by $|E^P|$, the rank of the service s_α in the recommendation list is dented by $p_{s\alpha}$, and k_{ux} is the number of services that are not rated/collected by the target user u_x in the training set. A smaller RS value depicts a greater recommendation accuracy.

Precision *(Jonathan et al., 2004)*

This metric evaluates the number of relevant services that a target user chooses to the total number of the top-K services that are recommended to the user. It is mathematically defined as Equation 9;

$$P_{ux}(K) = \frac{d_{ux}(K)}{K} \tag{9}$$

where $d_{ux}(K)$ represents the total number of relevant services the information gathered by the target user u_x

Recall *(Adomavicius & Tuzhilin, 2005)*

This is the number of relevant services selected by the target user to the total number of relevant services in the prob sets. Recall calculates the likelihood that a user will choose an appropriate service. Equation 10 uses it as a mathematical notation.;

$$R(K) = \frac{1}{m} \sum_{u_x=1}^{m} \frac{d_{u_x}(K)}{D(s_\alpha)} \tag{10}$$

where the number of all relevant services in the prob set is denoted by $D(s_\alpha)$. A greater precision and recall values indicate a higher recommendation accuracy. However, these two metrics often oppose each other, as $P(K)$ generally decrease with an increase in the value of K, whereas $R(K)$ normally increase with an increase in the value of K. To balance this trade-off, the F1 metric is introduced.

F1 Score *(Zeng et al., 2011)*

This is employed to balance the accuracy metric of both recall and precision, and is defined mathematically as Equation 11;

$$F1 = \frac{2 * P(K) * R(K)}{P(K) + R(K)} \tag{11}$$

The weights given to recall and precision are the same when using the F1 value as a combination metric. A higher F1 score represents a greater accuracy of recommendation and vice versa.

Inter-Similarity (IS)

The IS diversity metric (Zhou et al., 2009) looks at how the list of user recommendations is dissimilar from each other. For two users, u_i and u_j, the dissimilarity amongst their recommendation lists is estimated as in Equation 12;

$$IS_{ij}(K) = 1 - \frac{C_{ij}}{K} \tag{12}$$

where the number of similar services in the top-K places of the recommendation list of u_i and u_j is denoted by C_{ij}. Taking an average of $IS_{ij}(K)$ overall user pairs, we obtained the average inter-similarity score as shown in Equation 13. The higher the difference in users' recommendation lists, the greater $IS(K)$ score is.

$$IS(K) = \frac{1}{N(N-1)} \sum_{i \neq j} IS_{ij}(K) \tag{13}$$

***Novelty** (Zhou et al., 2009)*

Generally, for popular services, individual users can get their information easily from other channels, however, for unpopular services that are relevant, and typically more surprising and novel, it is challenging for users to get their information. The ability of recommendation methods to recommend unexpected and novel services can be calculated by a metric called novelty. Representing the top-K recommendation list of u_i by K_i, the novelty is mathematically defined in Equation 14 as:

$$Ne(K) = \frac{1}{NK} \sum_{i=1}^{N} \sum_{s_\alpha}^{K_i} K(s_\alpha) \tag{14}$$

A lower value of $Ne(K)$ signifies that a greater portion of the services recommended is unexpected and novel.

NUMERICAL RESULTS AND DISCUSSION

In this section, we present the experimental results. Table 3 (a, b, and c) shows the AUC, Ranking Score, precision, recall, F1, inter-similarity, and novelty scores of experiments performed on three different data sets. Table 3 (a, b, c) illustrates results compared under the different evaluation metrics on the datasets. As indicated in Table 3 (a), the suggested technique is more accurate when compared to other comparable algorithms using the Epinions dataset, Specifically, the proposed algorithm performs better on the AUC, Precision, Recall, and F1 score accuracy metrics than the base algorithms, except for the ranking score accuracy metric, where Trustwalker performs best. However, the diversity of items predicted is higher on MBD, while our proposed algorithm performs better on the novelty of predicted items. On the MovieLens-100K dataset, as shown in Table 3 (b), the proposed method performed better in all five accuracy metrics, the diversity metric, and the novelty metric. However, on the Netflix dataset, as shown in Table 3 (c), our proposed method performs better in four out of the five accuracy metrics and is slightly behind MBD in terms of novelty. The results show that the suggested technique is typically superior to the baseline methods in terms of recommendation accuracy. It also indicates that, as evaluated by the inter similarity and novelty scores in the three datasets, the proposed method tends to recommend novel items to the target user.

Table 3 a Performance Comparison of various recommendation methods on the Epinions dataset

Method	**Epinions**						
	AUC	RS	PR	RC	FI	IS	N
ListDb	0.821399	0.173025	0.023381	0.112419	0.038709	0.057536	253
MBD	0.822576	0.176421	0.023255	0.101907	0.037868	**0.035249**	175
TrustWalker	0.825485	**0.171487**	0.025374	0.116451	0.041669	0.056087	206
Proposed	**0.830859**	0.173297	**0.026058**	**0.124099**	**0.043072**	0.053195	**160**

Table 3 b Performance Comparison of various recommendation methods on the MovieLens-100K dataset

Method	**MovieLens-100K**						
	AUC	RS	PR	RC	FI	IS	N
ListDb	0.846521	0.089162	0.233144	0.202116	0.216524	0.2974312	305
MBD	0.863604	0.089282	0.252934	0.136951	0.177691	0.2613018	251
TrustWalker	0.889362	0.090297	0.238263	0.161886	0.192785	0.3017448	285
Proposed	**0.9011035**	**0.085264**	**0.270726**	**0.219474**	**0.242421**	**0.2501684**	**174**

Table 3 c Performance Comparison of various recommendation methods on the Netflix dataset

Method	**Netflix**						
	AUC	RS	PR	RC	FI	IS	N
ListDb	0.910452	0.062746	0.049263	0.406185	0.087869	**0.331726**	2014
MBD	0.912853	0.080172	0.046021	0.431847	0.098252	0.334284	**1629**
TrustWalker	0.908373	**0.060825**	0.040628	0.389254	0.092463	0.350148	2063
Proposed	**0.935285**	0.061072	**0.058124**	**0.479215**	**0.103673**	0.374823	1727

Furthermore, the performance of the proposed algorithm on experiments performed on the MovieLens-100 dataset, as presented in Table 3(b) is better than the base methods on all the evaluation metrics. Also, on the Netflix, dataset shown in Table 3(c), the proposed algorithm performs better in four of the five accuracy metrics (i.e. AUC, Precision, recall, and F1 – measure), while Trustwalker performs better on the Ranking score metric. In terms of the diversity of items recommended, ListDb is best while MDB is better on the novelty of recommendation of items

The robustness of the proposed method as measured by the F1-accuracy score is analysed through further experiments conducted by increasing the recommended list from 10 to 50, as illustrated in Figure 3. The findings demonstrate that, although the F1 score decreases as the number of items recommended rises, our proposed strategy outperforms the baseline methods in all datasets. This demonstrates that the proposed method is more robust in terms of the accuracy of prediction.

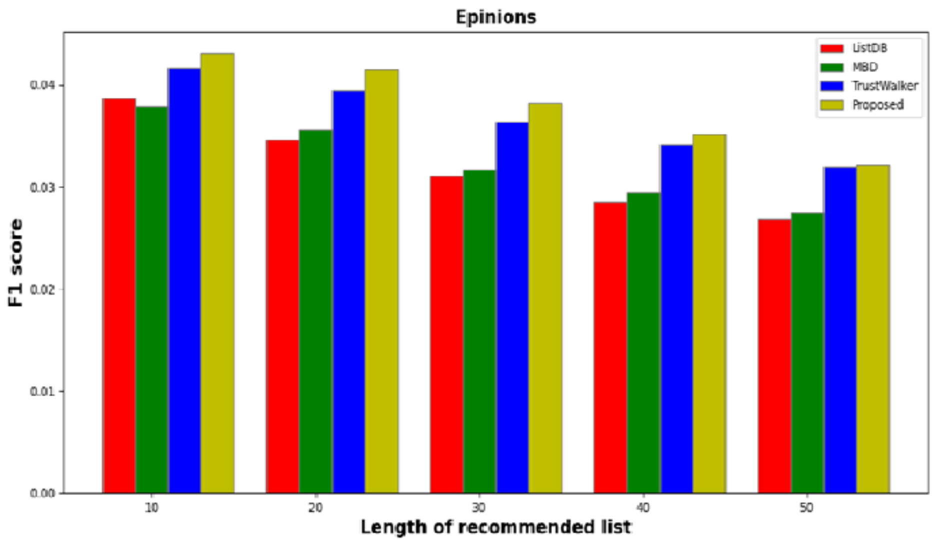

Figure 3a: F1- Score of algorithms on Epinions datasets with an increase in the size of recommended items

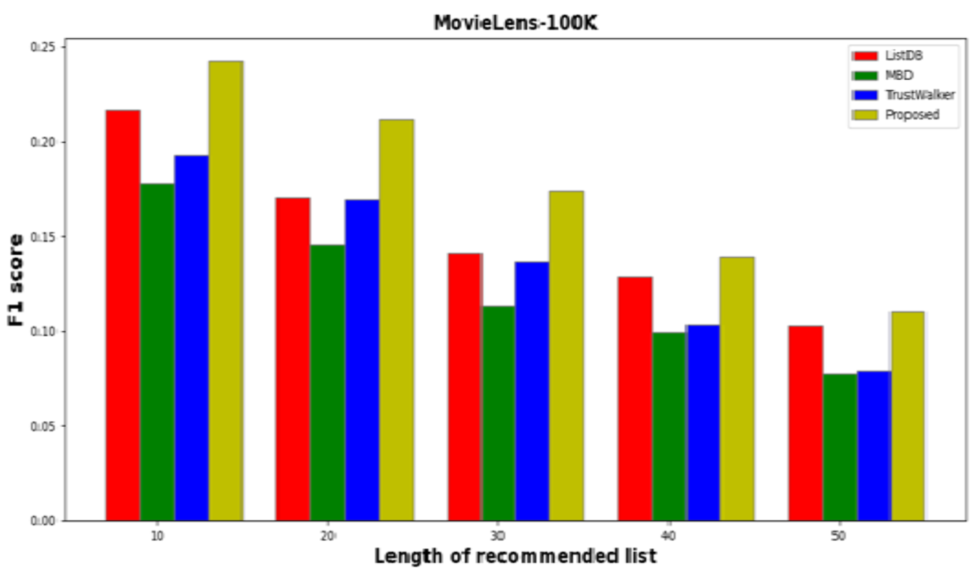

Figure 3b: F1- Score of algorithms on MovieLens-100k datasets with an increase in the size of recommended items

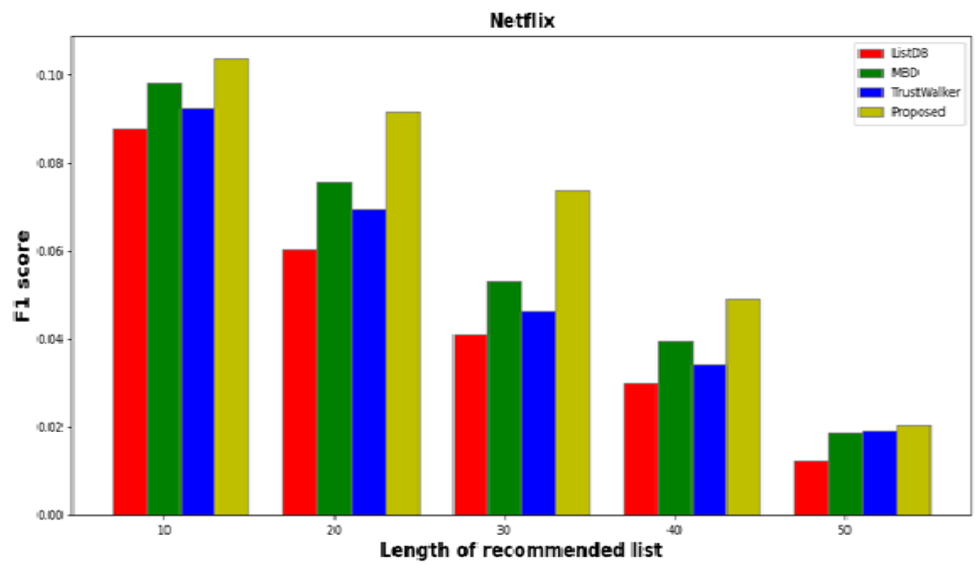

Figure 3c: F1- Score of algorithms on Netflix datasets with an increase in the size of recommended items

CONCLUSION

With the growth of the Internet, item recommendation has become very essential in recent years. It has therefore become necessary to adopt recommendation algorithms to recommend items to target users based on the behaviour of users. Many applications online adopt recommendation algorithms to gather information on the preferences of users and then predict the future interest of these users using the gathered information. To improve the recommendation accuracy as well as the diversity of recommendations, a new recommendation algorithm using the sigmoid function on the Jaccard similarity index is proposed. In the proposed method, we considered the rating significance of items and then used the sigmoid function on the Jaccard similarity index to estimate the asymmetric relationship between users. The resemblance between the target user and his or her neighbours is then assessed by combining the asymmetric relationship and the rating significance. Our model outperforms the baseline techniques in terms of item recommendation accuracy using AUC, recall, precision, ranking score, F1 accuracy metrics, and diversity using inter similarity and novelty metrics, according to several trials on three real-world datasets.

REFERENCES:

1.Aditya, P. H., Budi, I., & Munajat, Q. (2017). A comparative analysis of memory-based and model-based collaborative filtering on the implementation of recommender system for E-commerce in Indonesia: A case study PT X. *2016 International Conference on Advanced Computer Science and Information Systems, ICACSIS 2016*, 303–308. https://doi.org/10.1109/ICACSIS.2016.7872755

2.Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, *17*(6), 734–749. https://doi.org/10.1109/TKDE.2005.99

3.Alamdari, P. M., Navimipour, N. J., Hosseinzadeh, M., Safaei, A. A., & Darwesh, A. (2020). A Systematic Study on the Recommender Systems in the E-Commerce. *IEEE Access*, *8*, 115694–115716. https://doi.org/10.1109/ACCESS.2020.3002803

4.Aprilianti, M., Mahendra, R., & Budi, I. (2017). Implementation of weighted parallel hybrid recommender systems for e-commerce in Indonesia. *2016 International Conference on Advanced Computer Science and Information Systems, ICACSIS 2016*, 321–326. https://doi.org/10.1109/ICACSIS.2016.7872772

5. Campana, M. G., & Delmastro, F. (2017). Recommender Systems for Online and Mobile Social Networks: A survey. *Online Social Networks and Media*, *3–4*, 75–97. https://doi.org/10.1016/j.osnem.2017.10.005

6.Cao, Y., Li, W., & Zheng, D. (2019). A hybrid recommendation approach using LDA and probabilistic matrix factorization. *Cluster Computing*, *22*, 8811–8821. https://doi.org/10.1007/s10586-018-1972-y

7.Chen, G., Gao, T., Zhu, X., Tian, H., & Yang, Z. (2017). Personalized recommendation based on preferential bidirectional mass diffusion. *Physica A: Statistical Mechanics and Its Applications*, *469*, 397–404. https://doi.org/10.1016/j.physa.2016.11.091

8.Chen, Y., Wu, C., Xie, M., & Guo, X. (2011). Solving the sparsity problem in recommender systems using association retrieval. *Journal of Computers*, *6*(9), 1896–1902. https://doi.org/10.4304/jcp.6.9.1896-1902

9.Galeano, D., & Paccanaro, A. (2018). A Recommender System Approach for Predicting Drug Side Effects. *Proceedings of the International Joint Conference on Neural Networks*, *2018-July*, 1–8. https://doi.org/10.1109/IJCNN.2018.8489025

10. Hwangbo, H., Kim, Y. S., & Cha, K. J. (2018). Recommendation system development for fashion retail e-commerce. *Electronic Commerce Research and Applications*, *28*, 94–101. https://doi.org/10.1016/j.elerap.2018.01.012

11. Jiang, L., Cheng, Y., Yang, L., Li1, J., Yan, H., & Wang, · Xiaoqin. (2018). *jiang2018_ A trust-based collaborative filtering algorithm for E-commerce Recommender Systems.pdf* (pp. 3023–3034). Journal of Ambient Intelligence and Humanized Computing. https://doi.org/10.1007/s12652-018-0928-7

12.Jonathan, L. H., Joseph, A. K., Loren, G. T., & Riedl, J. T. (2004). Evaluating Collaborative Filtering Recommender Systems. *ACM Transactions on Information Systems*, *22*(1), 5–53.

13. Jothilakshmi, S., & Thangaraj, M. (2018). Design and Development of Recommender System for Target Marketing of Higher Education Institution Using EDM. *International Journal of Applied Engineering Research*, *13*(19), 14431–14437. http://www.ripublication.com

14. Katarya, R., & Verma, O. P. (2018). Efficient music recommender system using context graph and particle swarm. *Multimedia Tools and Applications*, *77*(2), 2673–2687. https://doi.org/10.1007/s11042-017-4447-x

15. Li, Y., Liu, J., & Ren, J. (2019). A Novel Learning Model Based on Trust Diffusion and Global Item for Recommender Systems. *IEEE Access*, *7*, 170270–170281. https://doi.org/10.1109/ACCESS.2019.2955863

16. Liu, H., Hu, Z., Mian, A., Tian, H., & Zhu, X. (2014). A new user similarity model to improve the accuracy of collaborative filtering. *Knowledge-Based Systems*, *56*, 156–166. https://doi.org/10.1016/j.knosys.2013.11.006

17.Mansoury, M., & Shajari, M. (2016). Improving recommender systems' performance on cold-start users and controversial items by a new similarity model. *International Journal of Web Information Systems*, *12*(2), 126–149. https://doi.org/10.1108/IJWIS-07-2015-0024

18.Massa, P., & Avesani, P. (2006). Trust-aware bootstrapping of recommender systems. *ECAI Workshop on Recommender Systems*, 29–33.

19. Mohamed, M. H., Khafagy, M. H., & Ibrahim, M. H. (2019). Recommender Systems Challenges and Solutions Survey. *Proceedings of 2019 International Conference on Innovative Trends in Computer Engineering, ITCE 2019, February*, 149–155. https://doi.org/10.1109/ITCE.2019.8646645

20.Patra, B. K., Launonen, R., Ollikainen, V., & Nandi, S. (2015). A new similarity measure using Bhattacharyya coefficient for collaborative filtering in sparse data. *Knowledge-Based Systems*, *82*(March), 163–177. https://doi.org/10.1016/j.knosys.2015.03.001

21.Saini, S., Saumya, S., & Singh, J. P. (2017). Sequential purchase recommendation system for E-commerce sites. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in*

Artificial Intelligence and Lecture Notes in Bioinformatics), *10244 LNCS*, 366–375. https://doi.org/10.1007/978-3-319-59105-6_31

22. Sanjeevan, A. Sadeghian, H. Rahnama, A. M. M. (2014). Recommender systems in E-Commerce. *IEEE*, 179–184. https://doi.org/10.4018/978-1-59140-562-7.ch073

23. Shoja, B. M., & Tabrizi, N. (2019). Customer Reviews Analysis with Deep Neural Networks for E-Commerce Recommender Systems. *IEEE Access*, *7*, 119121–119130. https://doi.org/10.1109/ACCESS.2019.2937518

24. Son, J., & Kim, S. B. (2017). Content-based filtering for recommendation systems using multiattribute networks. *Expert Systems with Applications*, *89*, 404–412. https://doi.org/10.1016/j.eswa.2017.08.008

25. Sun, C., Gao, R., & Xi, H. (2014). *Big Data Based Recommender System for non-E-commerce Retailers*.

26.Sun, L., & Xu, W. Z. (2016). E-commerce personalized recommendation system based on Web mining technology design and implementation. *Proceedings - 2015 International Conference on Intelligent Transportation, Big Data and Smart City, ICITBS 2015*, 347–350. https://doi.org/10.1109/ICITBS.2015.93

27. Tarus, J. K., Niu, Z., & Kalui, D. (2018). A hybrid recommender system for e-learning based on context awareness and sequential pattern mining. *Soft Computing*, *22*(8), 2449–2461. https://doi.org/10.1007/s00500-017-2720-6

28. Wang, Y., Deng, J., Gao, J., & Zhang, P. (2017). A hybrid user similarity model for collaborative filtering. *Information Sciences*, *418–419*, 102–118. https://doi.org/10.1016/j.ins.2017.08.008

29. Wenjun Li, Qiang Dong, Y. F. (2015). *List-wise Diffusion-based Recommender Algorithm from Implicit Feedback*.

30. Yadav, U., Duhan, N., & Bhatia, K. K. (2020). Dealing with Pure New User Cold-Start Problem in Recommendation System Based on Linked Open Data and Social Network Features. *Mobile Information Systems*, *2020*. https://doi.org/10.1155/2020/8912065

31. Yang, X., Guo, Y., Liu, Y., & Steck, H. (2014). A survey of collaborative filtering based social recommender systems. *Computer Communications*, *41*, 1–10. https://doi.org/10.1016/j.comcom.2013.06.009

32. Zeng, W., Zhu, Y. X., Lü, L., & Zhou, T. (2011). Negative ratings play a positive role in information filtering. *Physica A: Statistical Mechanics and Its Applications*, *390*(23–24), 4486–4493. https://doi.org/10.1016/j.physa.2011.07.005

33. Zhou, T., Ren, J., Medo, M., & Zhang, Y. C. (2007). Bipartite network projection and personal recommendation. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, *76*(4), 1–7. https://doi.org/10.1103/PhysRevE.76.046115

34. Zhou, T., Su, R. Q., Liu, R. R., Jiang, L. L., Wang, B. H., & Zhang, Y. C. (2009). Accurate and diverse recommendations via eliminating redundant correlations. *New Journal of Physics*, *11*. https://doi.org/10.1088/1367-2630/11/12/123008