A

Dissertation Report on

A Trust Based Collaborative Filtering Recommendation System using Deep Neural Network

Submitted

in partial fulfilment of the requirements for the degree of

Master of Technology

in

Computer Science & Engineering

by

Miss Hakke Ashwini Chandrakant

(Roll No. 2030008)

Under the Supervision of

Prof. Ajit Mali

DEPARTMENT OF COMPUTER ENGINEERING

K.E. Society's

Rajarambapu Institute of Technology, Rajaramnagar

(An Autonomous Institute, Affiliated to Shivaji University, Kolhapur)

2021-2022

K.E. Society's

Rajarambapu Institute of Technology, Rajaramnagar

(An Autonomous Institute, Affiliated to Shivaji University, Kolhapur)

CERTIFICATE

This is to certify that, Miss Hakke Ashwini Chandrakant (Roll No-2030008) has successfully completed the dissertation work and submitted dissertation report on "A Trust Based Collaborative Filtering Recommendation System using deep Neural Network " for the partial fulfillment of the requirement for the degree of Master of Technology in Computer Science and Engineering from the Department of Computer Science and Engineering, as per the rules and regulations of Rajarambapu Institute of Technology, Rajaramnagar, Dist: Sangli.

Date:

Place: RIT, Rajaramnagar

Dr.

Sign of Supervisor

Dr.	Dr. Sachin S. Patil
Sign of External Examiner	Sign of Head of Program
Dr Nagraj V. Dharwadkar	Dr. S. S. Gavade
Sign of Head of Department	Sign of PG Convener

DECLARATION

I declare that this report reflects my thoughts about the subject in my own words. I have sufficiently cited and referenced the original sources, referred or considered in this work. I have not misrepresented or fabricated or falsified any idea/data/fact/source in this my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute.

Place: RIT, Rajaramnagar Name of Student: Miss Hakke Ashwini Chandrakant

Date: Roll No: 2030008

ACKNOWLEDGEMENTS

I must mention several individuals and organizations that were of enormous help in the development of this work. I would like to express my sincere gratitude to my supervisor Prof Ajit Mali, for his guidance, nurturing, encouragement and support in every stage of my study. His knowledge, kindness, patience, open-mindedness, and vision have provided me with lifetime benefits.

I also very thankful to HoP, HoD for their valuable suggestions, critical examination of work during the progress, I am indebted to them.

In addition, very energetic and competitive atmosphere of the Computer Science and Engineering Department had much to do with this work. I acknowledge with thanks to faculty, teaching and non-teaching staff of the department, Central library and Colleagues.

I sincerely thank to Dr. Mrs. S. S. Kulkarni, for supporting me to do this work and I am very much obliged to her. Last but not the least I thankful to my parents who constantly supported me in all aspects.

Place: RIT, Rajaramnagar   Name of Student: Miss Hakke Ashwini Chandrakant
Date: Roll No:2030013

ABSTRACT

Recommender system (RS) are a type of suggestion to the information overload problem suffered by user of websites that allow the rating of particular item. RSs are one of the most successful and widespread applications of machine learning technologies in E-commerce. These techniques are used to predict the rating that one individual will give to an item or social entity. It uses the opinions of members of a community to help individuals in that community to identify the information most likely to be interesting to them or relevant to their needs. These systems use the similarity between the user and recommenders or between the items to form the recommendation list for the user. These preferences are being predicted using different approaches, namely content-based approach, collaborative filtering approach, etc. The movie RS are one of the most efficient, useful, and widespread applications for individual to watch movie with minimum decision time. Many attempts made by the researchers to solve these issues like watching movie, purchasing book etc., through RS, whereas most of the study fails to address cold start problem, data sparsity and malicious attacks.

To address data sparsity problem, we proposed Genre Similarity Based Trust Model which considers genre specific user similarity and permits only trusted similar profiles whose similarity exceeds with some predefined threshold value for each genre. And to solve the cold start problem, we proposed a new personalized recommendation model known as Genre Based Deep Collaborative Filtering (GDCF) which takes inputs as user item rating matrix and item genre as metadata for a better understanding of the latent features representation, which is usually not captured by the user item rating matrix. The performance evaluation of proposed model is estimated using Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE) on MovieLens dataset which shows a major improvement over the baseline methods with high accuracy of 81% with 0.60 MSE value.

Keywords: Collaborative Filtering, Deep Neural Network, Recommendation System, Trust.

Contents

List of Figures

List of Tables

ABBREVIATIONS

GDCF	Genre Based Deep Collaborative Filtering
CF	Collaborative Filtering
PCC	Pearson Correlation Coefficient

Chapter 1

Introduction

Many different domains have used recommender systems, including product suggestions, articles, book recommendations, movie recommendations, music recommendations, news recommendations, and web page and document recommendations. Recommender Systems (RSs) gather data on user preferences for a selection of goods (e.g., movies, jokes, applications, websites, travel destinations etc.) [19]. A user's behavior, such as the songs they listen to, the apps they download, the websites they visit, and the books they read, can be tracked explicitly (usually by collecting ratings given by users) or implicitly (often by tracking their online activity). RS may make use of user demographic information (like age, nationality, gender) In Web 2.0, social data like followers, followed, twits, and postings are frequently employed. Use of data from the Internet of things is becoming more prevalent (e.g., GPS locations, RFID etc.)[10]. Recommender systems are built using different methods. Depending on the information utilized to make recommendations, these systems are divided into Content based (CB) filtering or Collaborative Filtering (CF). CB techniques are based on profile of the target user's past interests the description of items. A CF technique is generally based on the notion that persons with similar preferences about certain items are likely to have alike preferences for other items [5].

Although Collaborative Filtering RS is the most popular recommendation system, it still has some flaws like data sparsity (If only small portion of items are rated by active users then it's difficult to find sufficient number of similar users), cold start problem (It represents the difficulty of making recommendations for new users and items), Scalability(If the volume of dataset is increased, then system is unable to generates satisfactory recommendations), and Graysheep problem (Gray sheep occurs in CF systems where the preferences and taste of a user do not equate with any group and ultimately, the recommendation process fails to yield good results)[8]. In actuality, comparing the ratings that two users provide for different items is a way to determine how comparable they are. The number of products that each user must rate in order for their ratings to be comparable is typically low, hence a minimum number of corated items must have been rated by both users.

Due to these limitations of CF recommendation system, trust and similarity between users are taken into account to raise the level of suggestion rather than only similarity-based users. Now a days, people prefer to acquire their information from people they can trust, like parents, friends, and relatives. The social network is a significant part of our everyday lives and has a wealth of information that may be used to enhance the recommendation system for better prediction. There are two types of trust among users. Implicit trusts describe the level of trust between users and can be determined using the data set's metadata. Explicit trust refers to the trust that the user expresses [17]. Obtaining explicit trust is challenging owing to data sparsity or lack of information. As a result, there are various metrics available that are used to determine implicit trust. This assessment pertains to how highly recommender systems value and consider the ratings offered by a particular user. The trust measure is employed as a connecting edge between two users, and this trust computation is readily represented in terms of a trust network. Looking for a collaborative filter and utilizing trust propagation across the trust network, one can create trustworthy users. The Trust-aware Recommender System recommends the items that these trustworthy users liked to the current user. By allowing users to express the degree of trustworthiness they place in one another, trust-based recommender systems have shown to be effective in addressing some of the drawbacks of CF-based techniques. In this study, Genre Similarity Based Trust Model which combines genre specific user similarity which shows two users to be highly similar when they like the same genre. A developed genre similarity based trust model that permits only trusted similar profiles whose similarity exceeds with some predefined threshold value.

1.1 Types of Recommendation System

The three primary classifications of recommendation systems are Content-Based Filtering (CBF), Collaborative Filtering (CF), Graph-Based Method, and Hybrid Recommendation.

1. **Content-Based Filtering (CBF)**:
   CBF is a traditional recommender system that suggests items with similar properties [6, 24, 25]. CBF is based on the item's description and takes into account previous user behavior. It generates a user's profile based on their history. Content-based recommender system utilizes the contents of items and finds the similarities among them.After analyzing sufficient numbers of items that one user has already shown favor to, the user interests profile is established. Then the recommender system could search the database and choose proper items according to this profile. The difficulty of these algorithms lies in how to find user preferences based on the contents of items. Many approaches have been developed to solve this problem in the areas of data mining or machine learning. For example, in order to recommend some articles to a specific reader, a recommender system firstly obtains all the books this reader has alreadyread and then analyzes their contents. Key words can be extracted from the text with the help of text mining methods, such as the well-known TF-IDF [8]. After integrating all the key words with their respective weights, a book can be represented by a multi-dimensional vector. Specific clustering algorithms can be implemented to find the centers of these vectors which represent the interests of this reader. For example, CBF extracts keywords from candidate papers, generates a user profile and calculates similarity [13]. High similarity scores are used to determine which papers are most similar to the user's search and which ones should be recommended to them. Webpages, news, and publishing recommendation systems all utilize CBF to a large extent.

2. **Collaborative filtering**:
   CF is widely used in movie and music recommendation. In CF, the preferences and interests of two or more similar users are taken into account. The collaborative filtering method collects and explores information about

users' past behaviour in terms of likes and dislikes. This includes monitoring the user's online behaviour and forecasting their interests by observing similarities of two users. In order to make recommendations for related products in the future, CF algorithms keep track of customer evaluations and ratings from the past. One such technique is matrix factorization. Unlike the content-based approaches, CF only relies on the item ratings from each user. It is based on the assumption that users who have rated the same items with similar ratings are likely to have similar preferences. CF is specifically designed to provide recommendations when detailed information about the users and items is inaccessible. Furthermore, it successfully mitigates the problem of over-specialization [10], which is quite common in content-based systems. Over-specialization is the phenomenon that recommended items are always much the same and the diversity of recommendations is neglected. As CF makes recommendations according to the neighborhood (people with similar preferences), the item one user has consumed may be something new tohis neighbors. The above features are particularly attractive which make CF algorithms extensively employed in recommender systems. For example, if the watch histories of users 1 and 2 are highly similar, it is likely that if user 1 watches a movie, user 2 will watch the same or something close to it [15].

3. **Hybrid recommendation**:
   hybrid recommendation system combines Content-Based filtering (CBF) and Collaborative filtering to suggest a broader range of items to users. The Hybrid recommendation system approach merges characteristics of multiple filtering approaches e.g. CBRS and CRS approach, by adding advantages of each approach there by achieving better performance [15]. This also alleviate various issues of RS such as ramp-up, data sparsity and overspecialization. HF approach is widely accepted and very helpful as it reduces the limitations of conventional recommender system. Firstly in 2002, author [17] suggested a taxonomy of seven hybridization techniques.Author [20] proposed a HRS algorithm by analyzing the similarity among content characteristics. This relationship is inserted in HRS to increase their accuracy. At first, a novel approach is used to obtain the content characteristic relation matrix. Then

the CRS is revised so that relation matrix can be efficiently merged with the algorithm. This algorithm alleviates the ramp-up problem. He divided hybridization technique into seven types. A hybrid recommendation system increases system performance by addressing the shortcomings of each separate algorithm.

4. **Demographics based recommender system(DBRS)**:
   DBRS works on the basis of user demographic data. The main goal of this RS approach is to classify the user on the basis of attributes and user's demographic data stored in their profiles (i.e. gender, age, location etc.) for suggesting the item. Unlike CRS and CBRS demographic approach doesn't need previous user ratings. DBRS works on three steps: Collecting input data, similarity computation and recommendation prediction. Collecting input data is the first step that includes new target user's demographic information (the user who wants recommendations) and also demographic information and rating of other users. Similarity Computation step uses user demographic information to retrieve users having related demographic interest thus creating a neighborhood. Ultimately, recommendation computation stage recommends items that are generally positively rated by surrounding users.Author proposed a DBRS for product recommendations which identify users buying plans from microblogs and make recommendations on the basis of matching user demographic data [17]. Conventional Product RS are generally designed for some particular e-commerce websites, which provides suggestions when users are carrying out some e-commerce activities.

1.2 Techniques to Design a Recommendation System

There are different techniques used to design a Collaborative Recommendation System. Collaborative filtering models which are based on assumption that people like things similar to other things they like, and things that are liked by other people with similar taste.

1. **Memory based approach**: Memory-based CF is one method that calculates the similarity between users or items using the user's previous data based on ranking. The main objective of this method is to describe the

degree of resemblance between users or objects and discover homogenous ratings to suggest the obscured items. The main idea of Memory-based CF is to recommend user items based on past ratings and reviews given by the user. This approach do remember the user's likings and disliking and based on it model gives personalized recommendations. In Memory Based Method the ratings are provided by users and the system compute the similarities between users and items. Based on all the ratings in database, the recommender system finds neighbours for certain user or item and calculates the predicted value for the unknown rating .These methods are simple to implement, easily understandable and do not require training phase[3]. The performance of memory-based CF is usually satisfactory on both accuracy and diversity. Memory-based approaches can be classified further into the following categories: User-based Approach and Item-based Approach.

In user-based CF technique, user profiling is done on the basis rating and feedback given or collected from the user and based on this profile, users having similar tastes are selected, and then they are given recommendations based on the similar users. In short it finds similarities between different users and using the rating from the top k closest (in similarity) users the system determines rating prediction or recommendation for the target user [9]. For each user, compute correlation with other users. For each item, aggregate the rating of the users highly correlated with each user [4]. These similar users are called neighbors and the scalability of these neighbors is a major issue for CF algorithms. An item based recommender find similarities between different items and the top k closest (in similarity) items are selected similar with the target item to determine rating prediction or recommendation for a particular user for a target item [9].

Unlike the user-based CF, item-based CF focuses on the similarities among items. It is based on the assumption that items with similar user ratings are likely to be of similar types. Hence the similarities among items are firstly calculated using the same similarity measures with the user-based CF. After choosing the neighbors for the target item and calculating the weighted average, the predicted rating on this item is obtained. It is easy to understand that once the items are too many and change frequently, scalability problem is also difficult to avoid.

2. **Model-based Approach**: Model-based Collaborative filtering approach involves building a model based on the dataset of ratings. In other words, we extract some information from the dataset, and use that as a "model" to make recommendations without having to use the complete dataset every time. This approach potentially offers the benefits of both speed and scalability.

   Although the memory-based CF is useful in effectively predicting missing ratings and presenting recommendations, it still has a few limitations. For instance, whenever a recommendation task is conducted, the system has to load all the ratings into the memory and implement specific algorithm based on the complete dataset. [10].Limited by the storage and computing resources, memory-based CF may often become quite time-consuming. Therefore, recommender system which can provide proper items with acceptable time consumption is highly desired .The model-based C.F. recommendation has been introduced to overcome the challenges of memory-based C.F. These techniques discover the rating pattern from historical data and give highly accurate and effective recommendations from some sample data.[8] The model-based RS requires a learning phase in advance for finding out the optimal model parameters before making a recommendation. Once the learning phase is finished, the model-based RS can predict the ratings of users very quickly. Clustering model and matrix factorization are familiar techniques used in model-based approaches. SVD, SVD++, and ALS are popular M.F. techniques that have gained importance in the Netflix price challenge's recommender system.

   Model-based collaborative filtering is not required to remember the based matrix. Instead, the machine models are used to forecast and calculate how a customer gives a rating to each product. These system algorithms are based on machine learning to predict unrated products by customer ratings. These algorithms are further divided into different subsets, i.e., Matrix factorization-based algorithms, deep learning methods, and clustering algorithms. Normally, the simple cluster algorithm is used like K-Nearest Neighbor to identify the nearest embedding or neighbor consisting of a similar matrix used for a product or a customer embedding. The matrix factorization technique is different from analyzing and exploring the rate of rating matrix in an algebra context and has two main goals. First, the initial ambition is to reduce the rating matrix dimension. This approach's second ambition is to identify perspective features under the rating matrix, which will provide several recommendations.

3. **Trust-aware Recommender System**: As another approach to alleviate the problems of traditional CF approaches (such as the data sparsity and cold-start), trust information has been widely used in CF methods.The resulting hybrid systems typically explore the trust network and find a neighborhood of users trusted (directly or indirectly) by a user and generate recommendations by aggregating their ratings [15]. As mentioned earlier, there are two main trust-based filtering methods: explicit trust and implicit trust. Since in explicit approaches trust values are obtained from pre-existing social links between users, asymmetry of the trust is always satisfied. In contrast, implicit approaches are often symmetric since they are based on the similarity or error measures which are symmetric in general [31]. According to [49], the performance of using implicit trust information is slightly worse than applying explicit trust information. Nevertheless, explicit trust-based filtering approaches encounter two major limitations [14]: (1) they require extra user efforts to label the trust statements. Accordingly, explicit trust statements may not always be available; (2) they suffer from the cold-start problem since new users should specify explicitly whom they trust before the filtering becomes effective. These limitations make the implicit trust-based filtering approaches more feasible to use [14]. Hence, the present work focuses on the implicit trust. With time, researchers have shifted their attention towards incorporating trust in RSs because of the problem of sparsity, malicious attacks etc. In traditional CF based RS, motivated by the fact the people generally agree upon the choices made by their friends, colleagues whom they trust. Trust information helps in overcoming the said problems.

   (a) **Requirement of Trust**: Trust plays an important role across many disciplines and forms an important feature of our everyday lives. In addition, trust is a property associated with people in the real world as well as social media. In recommender systems, it is defined based on the other user's ability to provide a valuable recommendation.

Concentration has been given towards incorporate trust in recommended system because of the problem of sparsity and malicious attacks. In traditional CF. Trust is an agreement between choices of the friends and colleagues upon the user. Evaluation of trust information tries to overcome the problems in CF. Implicit trust mined out from the similarity of the users while explicit trust value has been given by the user exclusively. Donovan and Smyth [4] discussed about the effectiveness of trusted users for recommendation. They proposed quality recommendations using Resnick formula on the basis of profile level and item level implicit trust.

i. **Types of Trust**:

Trust can be either of two types:

Explicit trust: Explicit trust is the trust value explicitly provided by the users.

Implicit trust: It refers to the trust information implicitly inferred from user behavior in the system e.g. user-item ratings.

ii. **Advantages of implicit trust over explicit trust:**

1) Explicit trust puts an extra burden on user for providing trust information apart form rating information.

2) Some popular datasets available for research purposes lack explicit trust information.

3) Most of the publicly available explicit trust data set contain trust scores in binary form.

4) Sometime explicit trust could be noisy. Users may trust each other due to various offline relations.

5) The amount of explicit trust information is relatively less than the number of ratings.

6) Implicit trust metrics are based on the instinct that users with similar ratings tend to be trustworthy.

There are limited studies on recommendation systems using Deep Neural Networks (DNNs), despite the fact that these technologies have had considerable success in a number of disciplines recently, including computer visionn, speech recognition, and natural language processing (NLP). In recent studies, some researchers presented deep learning-based recommendation algorithms, but most of these models use fixed interaction between user and items (user-item rating matrix) by neglecting additional information which leads to poor prediction accuracy. Therefore to consider extra information of an item we proposed an efficient Genre Based Deep Collaborative Filtering Model(GDCF) which takes Genre of the data as input along with user-item rating matrix that exploit item genre information to make more accurate recommendations. GDCF achieves much better performance with repeat to the conventional approaches.

The proposed system implements an efficient Genre Based Deep Collaborative Filtering Model (GDCF) which takes Genre of the data as input along with user-item rating matrix that exploit item genre information to make more accurate recommendations and Genre Similarity Based Trust Model which combines genre specific user similarity which shows two users to be highly similar when they like the same genre. A developed genre similarity based trust model that permits only trusted similar profiles whose similarity exceeds with some predefined threshold value. GDCF achieves much better performance with respect to the traditional approaches.

1.3 Research Challenges

The recommender system faces many challenges needed to be investigated and need solutions in this section we will introduce some of them.

1. Cold Start problem: cold start problem arises mainly when we have a new user to the site or when adding a new item to the system. Firstly, how we would recommend items to the new user we don't know his interests and he didn't rate any item yet. Secondly, to whom we can recommend this new item to others, even no one rate this item neither it's good or bad to be likely by users.

   The cold-start problem is also a serious concern similar to data sparsity issue. The active users with fewer feedback or the items with minimum ratings lead to cold-start problem. On the other hand, new users joined the system or newly added items suffer from the cold-start recommendation problem.

Due to cold-start problem and less amount of existing feedback information, the recommendation process lack in building suggestions based on similarity measures.

2. Sparsity Problem: Sparsity Problem is an important issue in recommender systems, it's happening when a user has large matrix contain buy items or watch movies or listing for music. Sparsity evolved when the user didn't rate these items. While recommender systems depend on users rating matrix users to recommend to the others.

   Though huge item ratings are available, it is known from the existing data that only a few users present their ratings as feedback for selected items. This circumstance leads to a low density of user feedback, approximately 1 percentage of the whole data. Thus the collaborative filtering algorithm faces several complications in rating prediction process due to the shortcomings of data sparsity or lack of ratings. Cosine similarity measures fade with its performance in identifying similar locations and users due to data sparsity problem.

3. Scalability: Scalability measure the ability of the system to work effectively with high performance while growing in the information. Recommender system needs to recommend items to the users without no change while the number of users increased or the number of items increased too. To achieve this we need more computations and get expensive.

4. Gray Sheep: gray sheep occurs in collaborative filtering systems where the opinions of a user do not equate with any group and consequently, is unable to obtain the benefit of recommendations.

1.4 Motivation of the Present Work

The main aim of A Trust Based Collaborative Filtering Recommendation System using Deep Neural Network to suggest or recommend movie to users. The most widely used recommender system is the collaborative RS. However it suffers from different problems such as cold-start, data sparsity, malicious attacks and Gray-sheep problem. Because of the restrictions and challenges of collaborative (CF) recommendation, instead of similarity based user, trust and resemblance between the users are taken into consideration for better recommendation. The proposed system implements an efficient Genre Based Deep Collaborative Filtering Model (GDCF) which takes Genre of the data as input along with user-item rating matrix that exploit item genre information to make more accurate recommendations and Genre Similarity Based Trust Model which combines genre specific user similarity which shows two users to be highly similar when they like the same genre. A developed genre similarity based trust model that permits only trusted similar profiles whose similarity exceeds with some predefined threshold value. GDCF achieves much better performance with respect to the traditional approaches.

1.5 Objectives of Present Work

1. To perform literature survey on trust based collaborative recommendation system and identify gap for new work.
2. To study and implement an existing trust based collaborative recommendation system using machine learning.
3. Proposing novelty in existing machine learning trust based collaborative recommendation system using deep neural network to achieve improvement.
4. To perform the performance comparison of existing machine learning trust based collaborative recommendation system and the proposed trust based collaborative recommendation system.

1.6 Layout of the Thesis

This work is arranged as follows: Chapter 1 gives brief overview of Recommendation system, its types, techniques, some research challenges and objectives of present work. In chapter 2 discuss about the history of the Recommendation system, literature survey related to deep learning based Recommendation system presents a literature review on Trustworthy CF recommendation systems . Chapter 3 discusses the theoretical concept of Deep Neural Network and trust based Recommendation System. Chapter 4 briefly discuss about methodology of proposed Recommendation system. Chapter 5 describes the comparative experimental results of DNN, DNN with Trust, GDCF and GDCF with Trust. Finally the chapter 6 describes the conclusion about proposed Recommendation system.

1.7 Closure

This section discusses the Recommendation system. Chapter 1 gives information about two characters of the Recommendation system and also gives an overview of some challenges faced while designing the Recommendation system. This section briefly discusses different Recommendation designing techniques. The research motivation and objective are also included in chapter 1.

Chapter 2

Literature Review

2.1 Introduction

For research purpose literature survey is very important part. It performs critical analysis on existing Recommendation system. Researchers can perform comparative study with the help of literature survey. It helps to find out gap or limitation of existing Recommendation system. It also helps to find out the problem statement. This Chapter covers different reviews carried out by the researchers for designing Recommendation framework to improve the presentation of the system. The performance of Recommendation system can be improved through different methodologies using collaborative filtering technique. It can be classified in two ways, Traditional CF Recommendation System and Trust Based CF Recommendation System by considering trust between two users.

Many different domains have used recommender systems, including product suggestions, articles, book recommendations, movie recommendations, music recommendations, news recommendations, and web page and document recommendations. Recommender Systems (RSs) gather data on user preferences for a selection of goods (e.g., movies, jokes, applications, websites, travel destinations etc.).

2.1.1 Traditional Collaborative Filtering Recommendation System

In contrast to the traditional clustering mechanism, the author has proposed a novel bio-inspired clustering technique that combines fuzzy clustering technique with swarm intelligence for UBCF. It has been evaluates on Yelp and Trip Advisor datasets for better recommendation accuracy [1].In order to improve the user similarity calculation, which is appropriate for sparse data and significantly boosts prediction accuracy, a new similarity measure approach is proposed in this study. This method takes user confidence and time context into account [2].The proposed approach considers a binary relationship between every single pair of item features in order to overcome the new item cold-start problem of IBCF. It also creates associated attributes to enhance the information about a new item and can be used to generate accurate and varied suggestions for new things in contemporary RS [3].In order to increase prediction accuracy, the author has presented a new MF model called ESVD (Enhanced SVD), which combines the traditional matrix factorization methods with ratings completion and the Multi-layer ESVD is used to manage the imbalanced data sets with more users than items or more users than items, respectively, the Itemwise ESVD and Userwise ESVD are introduced [4].The author has proposed a hybrid model that combines deep learning technology with a collaborative filtering recommendation algorithm. For latent features extraction, the model first applies a quadric polynomial regression model instead of the traditional MF techniques for feature representation. These latent features are then given to the deep neural network model as input data to predict the rating score. [5].A unique multi-criteria CF algorithm based on deep learning is proposed in this research. The model first acquires the features of the users and the items and feeds them to the deep neural network as an input for the criteria ratings prediction. Then the overall rating deep neural network to predict overall rating, which receives these criteria ratings as input [6].

The two classes of CF algorithms are memory based and model based algorithms. Recommendations based on complete user profile database known as Memory based algorithms while model based algorithms trained a model with previous history of user profile and recommend accordingly. Goldberg and Nichols [9] explains how collaborative filtering algorithms implemented in practical aspects to identify similar users in an information tapestry. Ungar and Foster [25] represents variations of K-means clustering and Gibbs Sampling on model parameters to estimate the model and compare with statistical model of collaborative filtering. Later, in Konstan et al. [13] automate the grouping of news articles in Usenet using CF too. Goldberg et al. [10] recommend jokes to the user and books by

Amazon using CF recommender systems based on Jester system. On the basis of user preference, CF system provides unexpected qualitative information to the user. But cold start, data Sparsity and Gray sheep problem is significant in CF systems. Accuracy of the CF system decreases due to data sparsity. Nagpal et al. [24] evade the data sparsity problem for feature selection of biomedical data using Gravitational search algorithm. Gupta and Hagpal [12] discussed when the taste of one user doesn't match with others, the CF system disable to produce recommendation in support to Gray sheep problem.

2.1.2 Trust Based Collaborative Filtering Recommendation System

A trust-based recommendation model is proposed for active user to select appropriate restaurants as per their choice which utilizes multicriteria ratings and implicit trust between customers to generate customized restaurant recommendations[7] .In this paper, For the selection of the nearest neighbour, the trust level is integrated. The expansion of various path lengths creates the trust network, and the trust transmission rules can be used to determine the trust value between users [8].Author has considered three aspect to improve the performance of proposed model, first to calculate the similarity between two items by considering common rated items then calculate the direct trust using trust weight and to calculate indirect trust between two users transfer mechanism is introduced. Finally user similarity score is combined with trust value and using trust weighting method prediction is generated [9].Author proposed trust-based collaborative filtering recommended system named as reputation-enhanced algorithm, which considers initial implicit reputation with explicit users response information to improve the performance of trust matrices and to alleviate the trust network sparsity and resulting into better recommendation accuracy[11].In this paper author implemented a new trust based filtering recommendation algorithm which uses Resnick's standard prediction formula .This filter allows only trustworthy profiles whose trust value exceeds some predefined threshold value using profile level trust approach and the model is evaluated on MoviLens Dataset[12].Author proposed new trust computation approach over traditional CF approach which considers similarity, centrality and social relationships. Probabilistic matrix factorization method is used for rating prediction of an item based on rating (user-item) matrix.

Similarity is calculated using Vector Space Similarity and Pearson Correlation Coefficient. Proposed trust model is validated on an Epinions dataset which shows better rating prediction by using similarity and centrality instead of using binary trust values [13].

Concentration has been given towards incorporate trust in recommended system because of the problem of sparsity and malicious attacks. In traditional CF. Trust is an agreement between choices of the friends and colleagues upon the user. Evaluation of trust information tries to overcome the problems in CF. Implicit trust mined out from the similarity of the users while explicit trust value has been given by the user exclusively. Donovan and Smyth [4] discussed about the effectiveness of trusted usersfor recommendation. They proposed quality recommendations using Resnick formula on the basis of profile level and item level implicit trust. Golbeck and Rovideskey [8] evaluates the trust value for a movie recommendation by using the Tidal Trust algorithm. By means of nearest trust path, this algorithm tries to find the scorings of all users made by the target user. Lathia et al. [14] proposed trusted k-nearest recommenders (KNR) algorithm to trust information between the users and tried to remove the drawbacks of traditional CF. Massa and Bhattacharjee [16] recommended common items using the explicit trust data from the users to conciliate the sparsity of the ratings. Massa and Avesani [15] proposed a novel approach to reduce the data sparsity and cold start problem for explicit recommendation using trust metrics and its propagation. Shamri and Bharadwaj [2] introduced different recommendation policies by exploring both trust and distrust. Computing weighted trust metrics and implementation of Fuzzy focused on to generate better recommendations. The above-mentioned algorithms use trust relationship among users and the similarity between users in addition it neglect the trust propagation mechanism between users, so it can't well solve the problem of data sparsity and Gray sheep problem. Keeping the problems occurred in CF, we considered both trust and similarity between users.

Many authors outperformed recommendations using deep neural network which takes input as user item rating matrix. A Genre Based Deep Collaborative Filtering (GDCF) which takes inputs as user item rating matrix and item genre as metadata for a better understanding of the latent features representation, which is usually not captured by the user item rating matrix and Genre Similarity Based

Trust Model which considers genre specific user similarity and permits only trusted similar profiles whose similarity exceeds with some predefined threshold value for each genre.

Chapter 3

Theoretical Background

This chapter will formally introduce some theory behind deep learning and trust computation techniques used in practice for designing proposed System.

3.1 Matrix Factorization

Matrix factorization is a way to generate latent features when multiplying two different kinds of entities. Application of matrix factorization [3, 23] in CF is to identify the relationship between items' and users' entities. With 95% sparsity in users' ratings, the recommendation model provides high computational complexity with low accuracy. Matrix factorization is a process, where the actual users' rating matrix is divided into two feature (user and item) matrices in such a way that multiplication of the new two matrices will give the actual matrix. The predicted matrix has similar output with the true values, and the 0 ratings in actual matrix are replaced with the prediction based on the similar users' preferences.Matrix factorization is easy to implement, potentially interpretable and requires less query time. However this cannot capture complex relation in data due to linear model.

3.2 Deep Neural Networks (DNN)

DNN is the multiple neural building blocks that can be composed in to a single differentiable and trained the network end to end. The interesting features are end to end differentiability and provide suitable inductive bias catered to the input data. Data complexity and large training samples leads to reasonable performance gain.

The deep learning method tried to predict the rating of an item for an active user and recommend an item to a user accordingly. The embedding layer receives the user-item feature matrix as an input. The dot product of two feature matrices are passed through dense hidden layer in which,number of nodes in the layer is equal to the number of features. Accuracy of the model is measured through MSE (Mean square error).

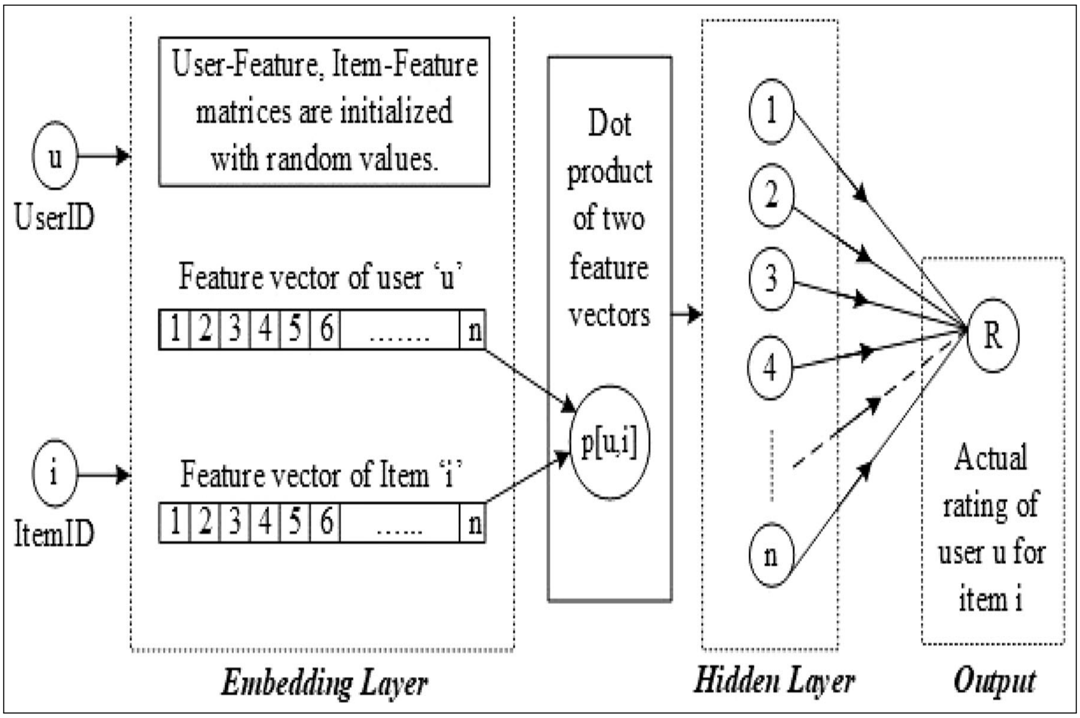

Figure 3.1: Deep Neural Model Architecture

The deep learning method tried to predict the rating of an item for an active user and recommend an item to a user accordingly. The embedding layer receives the user-item feature matrix as an input. The dot product of two feature matrices are passed through dense hidden layer in which, number of nodes in the layer is equal to the number of features.

3.3 Trust Based Filter

In CF, Consumer searches for the information and producer provides the information. The technical hitches of CF systems can be dealt with trust properties. The trust is assessed as readiness to accept the truth in a client on the basis of expertise and performance in a particular time. A trust system is considered as a network of interfacing peers. The communication between two domestic entities is considered as an end product of building the trust association with an additional entity by the prescribed representations of trust. According to Donovan and Smyth Local trust and Global trust are two important distinction of trust metric. Without peer dependency, computation of a single score termed as Global Trust. Local trust metric provides personalized scores, means it suggests trustable peer. Predicted rating for an item i and consumer c, can be evaluated based upon producer profiles, associated with their individual average recommendations according to similarity with the consumer. Resnick's standard prediction rating will be

$$c(i) = \overline{c} + \frac{\sum_{p \in P} T_{(i)}(p(i) - \overline{p}) * sim(c,p)}{\sum_{p \in P} T_{(i)}|sim(c,p)|} \tag{3.1}$$

For consumer profile c, c(i) is the rating to be predicted for the item i and for producer profile p, p(i) is the rating for the item i. c and p are the mean ratings of c and p respectively. The similarity between profiles c and p were measured by Pearson's correlation coefficients. The contribution to the rating prediction of a partner to its target user depends on the level of similarity between them given by Resnick. High similarity contributes more to the system in prediction of rating. Trust p, the profile-level trust is the percentage of correct recommendations that the producer has contributed.

$$Trust^p = \frac{|CorrectSet(p)|}{|RecSet(p)|} \tag{3.2}$$

Where correct recommendation set is

$$Correct(i,p,c) \approx |p(i) - C(i)| < €$$

The difference of rating greater than some threshold value will be correct (i, p, c) recommendation and the whole recommendations for a producer is RecSet(p). Trusted based recommendation is filter decides the trustworthy profiles to participate in the prediction process. The modified version of Resnick's formula allows producer profiles whose trust value exceeds some predefined threshold to participate in the recommendation process.

The standard Resnick method applied to the most trustworthy profiles given (Eq. 3.3).

$$C(i) = \overline{C} + \frac{\sum_{p \in P} T_{(i)}(p(i) - \overline{p}) * sim(c,p)}{\sum_{p \in P} T_{(i)}|sim(c,p)|} \tag{3.3}$$

$$P^T(i) = \{p \in P(i) : Trust^P(p, i)\}$$

In a recommender system, trust values for producers can be annulled from the similarity measure between users and the variation of predicted rating and the actual rating.In the proposed model an active user is being tested for cold user. Not being a cold user, trust between the intended users with other users in the system was calculated using Donovan's trust formula. Performance of all referred models being calculated and combined with trust value produced for the recommendation.

3.4 Popularity Based Recommendation System

In the proposed model an active user is being tested for cold user. Not being a cold user, trust between the intended users with other users in the system was calculated using Donovan's trust formula. Performance of all referred models being calculated and combined with trust value produced for the recommendation. For a cold user, Optimal Score Generator generates a score based on the rated movies and sorted in a descending order.

$$Score = (timestamp/max(timestamp)) * z_score \tag{3.4}$$

High order optimal score along with user preferences recommends the exact movie.

Chapter 4

Methodology of Present Work

4.1 Introduction

This chapter discusses about proposed architecture and mathematical model of GDCF Model. Following that, the Genre Similarity Based Trust Model is discussed using PCC proposed by Manos Papagelis[14]. This chapter gives detailed explanation of proposed Recommendation system.

4.2 Overview of proposed System

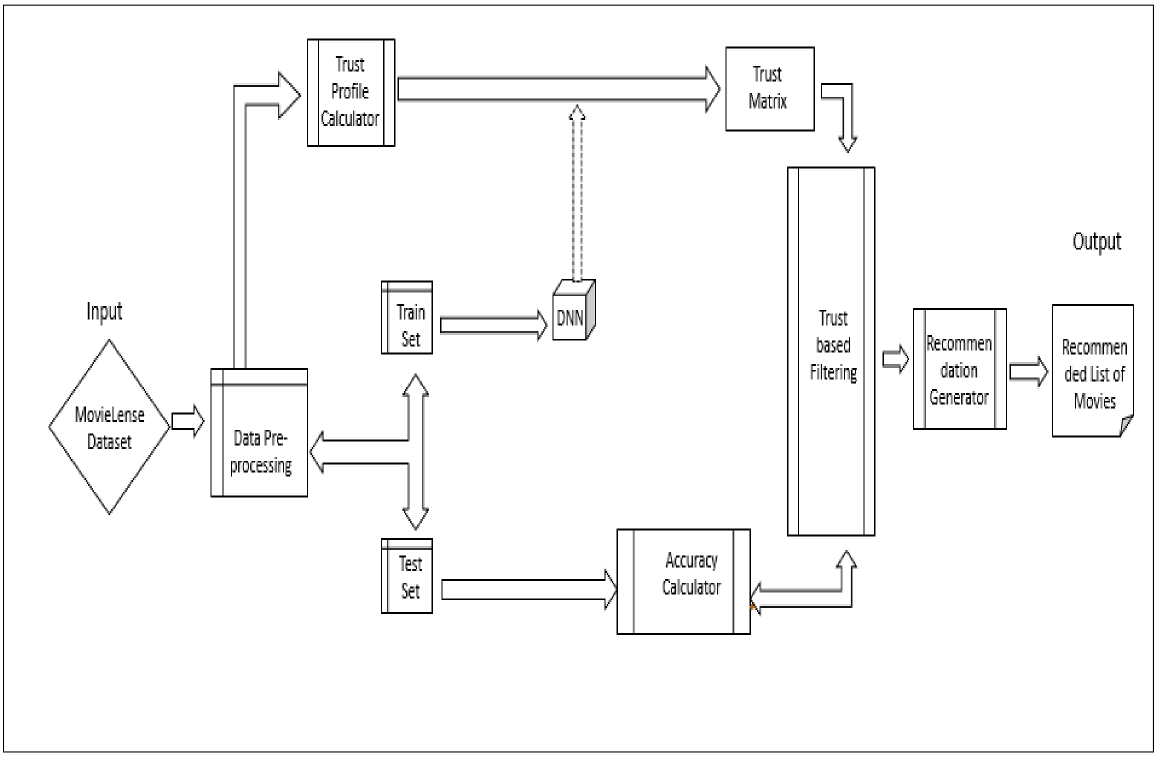

Figure 4.1: Proposed Model Architecture

As shown in Fig 4.1, the proposed system architecture which consists of two modules named as GDCF and GDCF with Trust. Initially, data is preprocessed and split up into training and testing set. In the GDCF module, to acquire the latent features of the users and items, we use user item interaction rating matrix and Genre of item to obtain the latent features. Then these latent features are given as input to GDCF model which uses the forward propagation algorithm. We will receive some probability values that indicate the predicted rating in the output layer. Finally, the highest probability score will be used as the prediction result for a target user and accuracy is calculated through Mean Squared Error (MSE) on Test Set.

On the Other hand In terms of the particular application area, we view user associations as a manifestation of mutual trust that has been built up through time. The more alike two users are, the more their formed relationship of trust would be regarded, as trust is defined in terms of similarity conditions. In proposed Trust based GDCF Model, we propose Genre Similarity Based Trust method which combines genre specific user similarity. In this, two users to be highly similar when they like the same genre and to calculate the similarity between two users, Pearson Correlation Coefficient (PCC) proposed by Manos Papagelis[14]. A developed trust computational model that permits only trusted similar profiles whose similarity exceeds with some predefined threshold value and then incorporate trust inference in GDCF systems by modifying the predicted ratings of specific items as per trusted users rating. Performance of GDCF model being calculated and combined with trust value produced for the recommendation.

4.2.1 Genre Based Deep Collaborative Filtering Neural Network (GDCF)

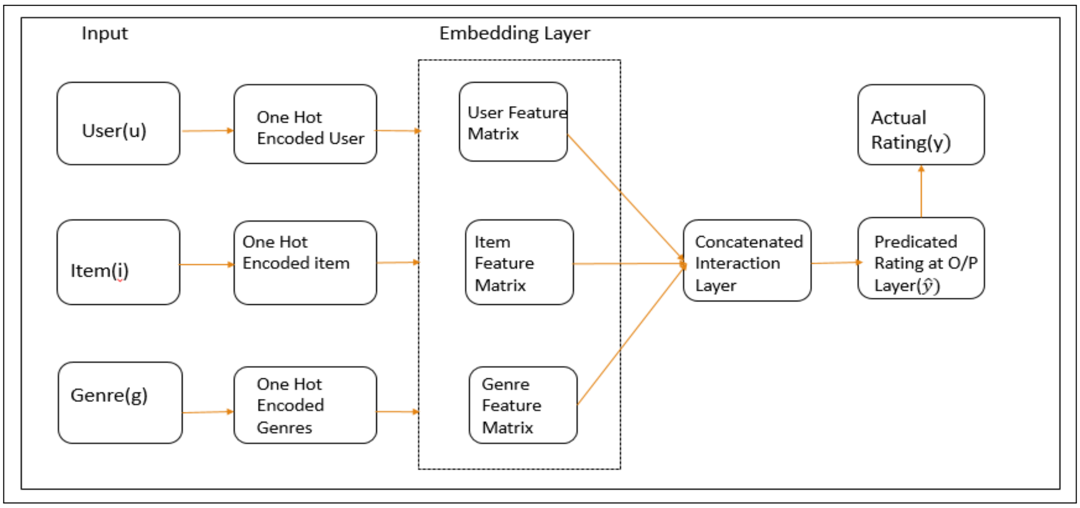

Figure 4.2: Genre Based Deep Collaborative Filtering Neural Network

The GDCF Neural Network model is employed as the neural network architecture to build a user item prediction model and to benefit from fully connected neural network based CF where the neural network's next layer is fed the output of the previous layer, as shown in Fig 4.2.For utilizing metadata features of user and item, there are number of ways and applications to customize Input Layer. In our system, we have considered genre of the movie for user item interaction that exploit item genre information to make more accurate recommendations. The collaborative framework covered by NCF [18] uses the one-hot encoded binary values as the user and an item input features. It is the most general and uncomplicated form that is adaptable to any input feature vector. To overcome the cold start issue, the one-hot encoded vector demonstration is modified to contain metadata information of user and item. The content features are deemed to be the user's demographic data as well as the genre of item. The user-item metadata information's binarized values are represented by the input user-item feature vector. In this model the input layer of the network is an embedding layer that is fully connected layer that transforms the sparse vector representation that was one-hot encoded into a dense vector representation. The latent features are represented by these embedded vectors of user, item and genre etc then the neural network architecture concatenates all latent features at hidden layer. It converts these features into extra low dimensional latent vectors to help to understand the complicated non-linear relationship between the item and user metadata. It is possible

to modify each layer to state a more complex hidden latent feature interaction. The competency of model is determined by the final hidden layer dimension.
As discussed above GDCF Model in figure 4.2, which takes the latent features of users items and genre of the item as the inputs and uses the forward propagation algorithm to predict the rating scores. According to our model, in the input layer, the input vector x0 is concatenated by the latent features of users and items; therefore, for any record R_{ij}, we have

$$X_0 = concatenate(U_i, V_j) \tag{4.1}$$

Where the function concatenate() is used to concatenate two vectors. When X_0 passes through the first hidden layer, the output of the first hidden layer is obtained by the following equation:

$$X_1 = activation(W_{1x_0} + b_1) \tag{4.2}$$

where W_1 is the weight matrix between the input layer and the first hidden layer, b_1 is the bias vector, the activation() indicates the activation function, which is designed to make the neural network model nonlinear and multilayer neural networks become meaningful. In the DNN model, the activation functions that we use include the sigmoid, tanh, and ReLU functions. In this paper, we choose ReLU as the activation function for our model because it is more effective and easier to optimize [26].
By Eq. (4.2) and the discussion above, we can obtain the output at the l-th hidden layer:

$$X_1 = ReLU(W_{lx_{l-1}} + b_1) \tag{4.3}$$

In the output layer, our training goal is to predict the user's rating score R_{ij}. We use the One-Hot encoding method again to obtain the supervised value y =OneHot(R_{ij}); therefore, we need to transform the output result by the softmax method to obtain the prediction value of the corresponding position of y, that is

$$y = softmax(W_{out}x_h + b_{out}) \tag{4.4}$$

where h represents the number of hidden layers, x_h is the output of the last hidden layer, and W_{out} and bout represent the weight and bias of the output layer, respectively.

4.2.2 Genre Similarity Based Trust Model

Building a user-item rating matrix and computing user similarity are essential components of collaborative filtering. Collaborative filtering techniques, however, have a number of drawbacks, including cold start users and data sparsity. Few research studies have included user similarity in trust models as a solution to this problem. Reconstructing the trust matrix with the use of weighted trust propagation and user similarity aids in solving the cold start issue. The authors suggested an algorithm for trust score that builds a trust relationship matrix by combining the quantity of items and the similarity score between users. Another study [16] presented a trust model based on the similarity of user rating behaviors and propagated trust. According to Donvaon and Smyth [17], there are two key variations of trust metrics: local trust and global trust. Without peer dependence, a single score called “Global Trust” is computed. Local trust metrics offer modified rankings, indicating trustworthy peers. Based on producer profiles and their unique average recommendations in relation to similarities with the consumer, predicted ratings for an item and consumer can be assessed.

The above trust Model is solely based on the concept if producer profiles has performed well before in terms of rating prediction, then these profiles can be considered as more trustworthy than one whose predictions capability is poor. In each recommendation session, the producer profiles generate the recommendations and consumer profiles receives the recommendation. This trust based Recommendation System filters the trustworthy Producer profiles whose trust value exceeds a certain threshold are included to participate in the prediction process. Making recommendation using Donovon's trust formula is two-step practice.

In order to calculate the trust score for any given pair of users, pairs rating vectors must first be produced. This process results in the creation of a trust matrix. Making preliminary prediction and determining similarity are further divisions of this process. The final predictions are given by substituting a trust matrix for the similarity matrix. The method's computing cost is significantly increased by the creation of the trust matrix.

According to our methodology, user similarity and relative importance each have an impact on user trust, therefore we have considered similarity between two users to calculate the trust.

In Genre Similarity based Trust Model, first we considered user preference for item genre and classify users according to their preference for item genres. Selecting an active user of a particular genre means user prefer that genre therefore build a genre rating $< u\_g >$ matrix from the master dataset, including user id, item id and item genre (u-id, i-id, r, i-g) by calculating the user preference average per item genre (g), which shows two users to be highly similar when they like the same genre. In second step Similarity between users' preference was determined according to the average rating for the item genres using below equation

$$diff(a, u_i) = |avr\ r_{a,gj} - avr_{r\ ui,gj} \tag{4.5}$$

a, and ui preferred gj

where i=1...n, and j=1...,s.

$diff(a, ui)$) represents the degree of difference between a and u preferences for the same genre g. Smaller the value, more the similarity between users. $avr\ r_{a,gj}$ represents the average ratings of a to $gj.avr_{r\ ui,gj}$ represents the average ratings of ui to gj.

Then calculate the trust between the target users with other users in the system using Trust Inference proposed by Manos Papagelis which uses Pearson Correlation Coefficient (PCC) in equation 3.

$$sim(u_x, u_y) = \frac{\sum_{h=1}^{n'}(r_{u_x}, i_h - \overline{r}_{u_x})(r_{u_y}, i_h - \overline{r}_{u_y})}{\sqrt{\sum_{h=1}^{n'}(r_{u_x}, i_h - \overline{r}_{u_x})^2}\sqrt{\sum_{h=1}^{n'}(r_{u_y}, i_h - \overline{r}_{u_y})^2}} \tag{4.6}$$

Where i=1...n(numbers of item)

r_{u_x}, i_h as the rating of user u_x to item i_h and

r_{u_x} and r_{u_y}, are the average ratings of users u_x and u_y respectively, then the trust between two users is defined as the Pearson correlation [20] of their associated rows in the user-item matrix. A trust model has been developed that permits only trusted similar profiles denoted by V^T using equation (4) whose similarity exceeds with some predefined threshold value and then incorporate trust inference in GDCF systems by modifying the predicted ratings of specific items as per trusted users rating.

$$V^T(i) = v \in V(i)Trust^P(v,i) > T \tag{4.7}$$

The key components of the given model are to first train the GDCF model using the training data that is provided, and then modify the predicted rating using predicted ratings from users in the user's trusted user network. Equation 5 is used to calculate the adjusted rating value. The weighted sum of ratings from trusted users and the user's average rating make up the modified rating, which is divided into two parts.

$$R_{i,k} = R_i \pm \frac{\sum_{v \in V}(F_{v,k} - F_v) * T(i,v)}{\sum_{v \in V} T(i,v)} \tag{4.8}$$

where Ri is the user i's average rating, V is the user i's trusted user set, and Fv is the user v's average rating from the trusted user set v. The performance of the suggested method is to be assessed using the evaluation metric of actual ratings and changed predicted ratings once the modified predicted ratings are computed. Performance of the DNN model is calculated, and the recommendation's trust value is coupled with it.

4.3 Closure

In this chapter proposed methodology explained. The architecture and mathematical structure of proposed model described briefly in this chapter.

Chapter 5

Experimental Environment

5.1 Introduction

This chapter discusses experimental environment such as software requirements for proposed Recommendation system, dataset and evaluation matrices used to measure the performance of the proposed system. This chapter gives information about experimental results and analysis.]. A developed trust computational model that permits only trusted similar profiles whose similarity exceeds with some predefined threshold value and then incorporate trust inference in GDCF systems by modifying the predicted ratings of specific items as per trusted users rating. Performance of GDCF model being calculated and combined with trust value produced for the recommendation.

5.2 Experimental Environment

1. Operating System: For proposed Recommendation system Windows operating system is used.

2. Programming Language: Python programming language is used to implement proposed Recommendation system. This language contains some build-in packages for machine learning. For implementation purpose python 3.5 version is used.

3. Keras: Keras is a deep learning API written in Python, running on top of the machine learning platform Tensor Flow. It was developed with a focus on enabling fast experimentation. Being able to go from idea to result as fast

as possible is key to doing good research.

4. TensorFlow: TensorFlow is a software library or framework, designed by the Google team to implement machine learning and deep learning concepts in the easiest manner. It combines the computational algebra of optimization techniques for easy calculation of many mathematical expressions.It includes a feature of that defines, optimizes and calculates mathematical expressions easily with the help of multi-dimensional arrays called tensors. It includes a programming support of deep neural networks and machine learning techniques.

5. Google Colabaratory: For faster training the proposed system requires GPU. The Google Colaboratory provides free GPU services for limited period of time. Colab is a Python development environment that runs in the browser using Google Cloud

5.2.1 Dataset

The dataset constraint in our experiment is the dataset in the proposed algorithm does not contain only rating data but also information about the genre and the user profile data (namely age, gender, occupation, and location) to capture user's preferences. The statistics of the ml –latest-small dataset is shown in Table 5.1.

Table 5.1: Genre name in ml-latest-small dataset and number of movies per genre

No	gName	No.of Movies	No.	gName	No.of Movies
1	Drama	13,344	11	Mystery	1514
2	Comedy	8374	12	Fantasy	1412
3	Thriller	4178	13	War	1194
4	Romance	4127	14	Children	1139
5	Action	3520	15	Musical	1036
6	Crime	2838	16	Animation	1027
7	Horror	2611	17	Western	676
8	Documentary	2471	18	Film -Noir	330
9	Adventure	2328	19	No Genre Listed	246
10	Sci-Fi	1743			

This ml-latest-small dataset [MovieLens] is used for experimentation (http://

movielens.org). These data were created by 610 number of users between 29 March 1996 and 24September 2018 includes 100837 ratings for 9741 number of movies. The included users were randomly selected and each selected user had given rating to at least 20 movies. The dataset files are presented as comma-separated values which includes 'links.csv', 'movies.csv', 'ratings.csv' and 'tags.csv'.For experimentation purpose, we have considered only movies.csv and rating.csv
'rating.csv' file contains rating for all users in the form of userId, movieId, rating, timestamp 'movies.csv file contains Movie information in the form of movieId, title, genres.
Genres of movies are pipe separated list which includes Action, Adventure, Animation, Children's, Comedy, Crime, Documentary, Drama, Fantasy, Film-Noir, Horror, Musical, Mystery, Romance etc

5.2.2 Evaluation metrics

Predictive accuracy measures such as MAE, RMSE are used for performance evaluation purposes. To analyse the performance of our model, the Root Mean Square Error (RMSE) is selected to measure the loss in the prediction system.

Root Mean Squared Error (RMSE):

Root Mean Square Error (RMSE) is the standard deviation of the residuals (prediction errors). Residuals are a measure of how far from the regression line data points are; RMSE is a measure of how spread out these residuals are. In other words, it tells you how concentrated the data is around the line of best fit. Root mean square error is commonly used in climatology, forecasting, and regression analysis to verify experimental results.
The RMSE can properly indicate the accuracy of the measurement because it is accurate to both very large and extremely less errors in the measurement set. The RMSE is defined as

$$RMSE = \frac{\sqrt{\sum_{i=1}^{N}(Predi - Acti)}}{N} \tag{5.1}$$

where N denotes the total number of items in testing set. Acti shows the actual rating of user i and Predi shows the predicted rating of user i.

Mean Absolute Error

In statistics, mean absolute error (MAE) is a measure of errors between paired

observations expressing the same phenomenon. Examples of Y versus X include comparisons of predicted versus observed, subsequent time versus initial time, and one technique of measurement versus an alternative technique of measurement. MAE is calculated as the sum of absolute errors divided by the sample size.
The average of all absolute errors is known as the mean absolute error (MAE), whereas the absolute error is the degree of mistake in your measurements.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|x_i - x| \tag{5.2}$$

Where n = the number of errors,
$\sum$= summation of all absolute errors
$|x_i - x|$ = the absolute error

5.3 Experimental Work

5.3.1 Data Preprocessing

The first step is data preparation to perform data pre-processing by reducing irrelevant attributes. These irrelevant attributes are the timestamp of the rating data, movie title, release date, video release date, and IMDb URL of the item data.

For a real-time system scenario, cleaning the dataset is a crucial step that involves extractingthe relevant features from the dataset and encode these features in integer values forfurther processing in neural network architectures. The MovieLens datasets are cleaned byremoving irrelevant features such as timestamp, age, zip code, movie title, release date,video release date, and IMDb URL. The gender and occupation features as the user's metadataand genre feature as the item's metadata are adopted since both are helpful in betterrating prediction. All the relevant features are binary encoded using categorical encodingprovided by Pandas. The Keras embedding layer is applied to the binary encoded features which learn embeddings of the user-item metadata information.

5.3.2 Algorithm

A) Generate Prediction

- STEP: 1: Reading Data and the pre analysis

- STEP: 2: Movies table with genres in one hot format
- STEP: 3: Apply DNN on UserId , movieId and Genres
- STEP: 4 : Generate Prediction

B) Generate Recommendation

- STEP: 1: Building a genre rating ¡u g¿ matrix from the master dataset, including(u-id, i-id, r, i-g) by calculating the user preference average per item genre (g).
- STEP: 2: Finding the similarity between the users' preferences per genre using Pearson Coefficient Correlation.
- STEP: 3: Finding the nearest neighbors (K-NN) for each user per genre by arranging the difference values in the preferences rate in ascending order.
- STEP: 4: Generate Recommendation

5.3.3 Experimental Results

In this section all the outputs and results of the above implemented model and comparison of their validation loss vs epoch were given.

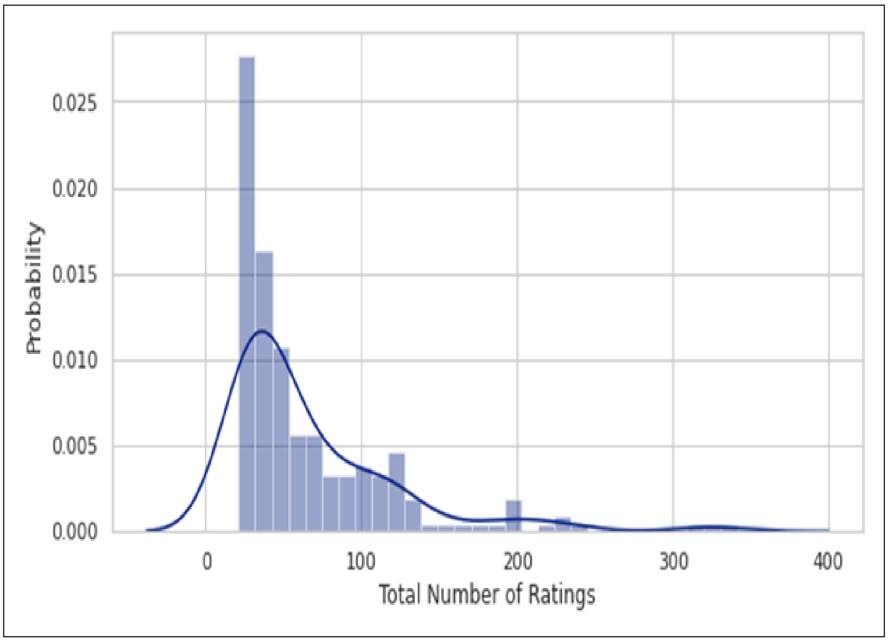

Figure 5.1: Probability vs Total No of Ratings

The input dataset for this model consist of 610 number of users for 9724 number of movies with 100,837 ratings. Probability of movie ratings is plotted against total no of ratings (see Fig 5.1).

1. DNN with user item interaction:

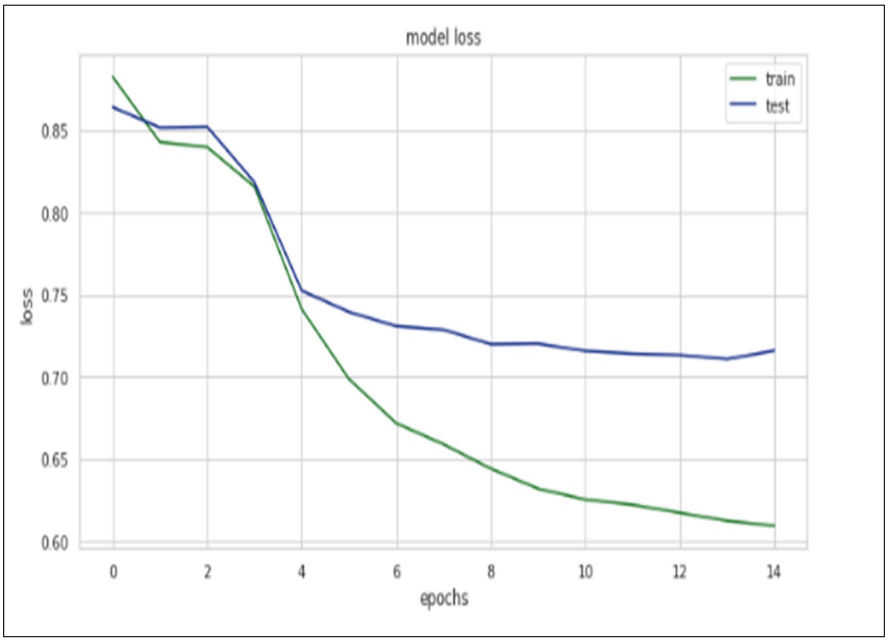

Figure 5.2: Validation Loss of DNN Model

The input dataset for this model consist of 610 number of users for 9724 number of movies with 100,837 ratings.The user-item feature matrix is fed into the embedding layer. The concatenated product of two feature matrices then given to dense hidden layer for generating rating prediction (see Fig 5.2).

3. Movie Genre Data Visualisation:

Figure 5.3: Movie Genre Visualisation

Genres of movies are pipe separated list which includes Action, Adventure, Animation, Children's, Comedy, Crime, Documentary, Drama, Fantasy, Film-Noir, Horror, Musical, Mystery, Romance etc. There are total 26 types of genres are considered to carry out this experimentation as shown in Fig 5.3.

4 DNN with Metadata (Genre)

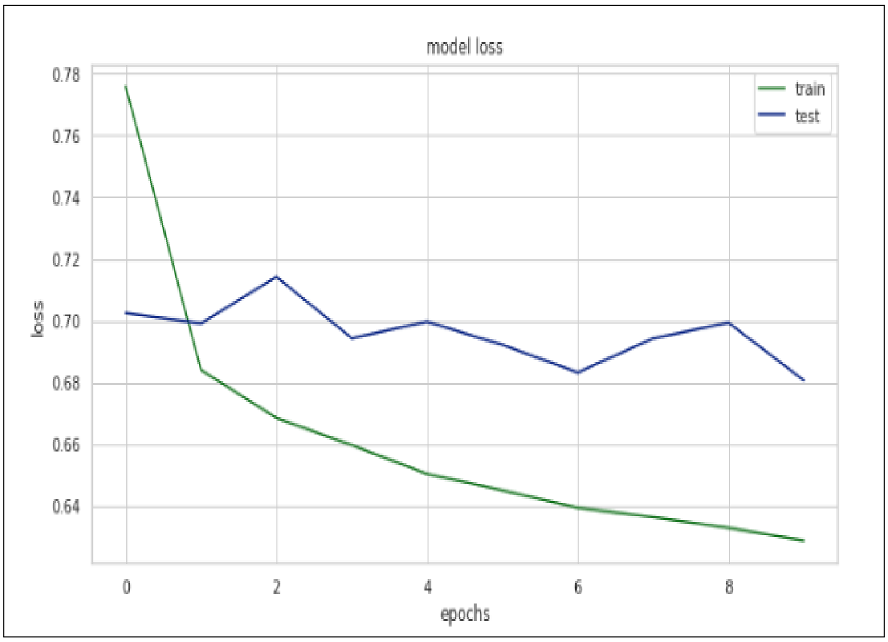

Figure 5.4: Validation Loss of DNN with Metadata (Genre) Model

Along with user item rating matrix, genre of the movie is given as input to embedding layer for better learning of user feature matrices. Three user feature matrices are then concatenated and input to neural network for rating prediction. To calculate the overall performance of proposed model, Mean Square Error and Mean Absolute Square Method is used which shows significant improvement on baseline methods. Training loss and validation loss is shown in Fig 5.4

5 Genre Based Similarity Calculation:

Genre rating $< u\_g >$ matrix is generated from the master dataset, which includes user id, item id and genre of the item through which average genre preference is calculated as shown in Fig 5.5

UserID	1	2	3	4	5	6	7	8	9	10	...	601	602	603	604	605	606	607	608	609	610
1	1.000000	0.469994	-0.550472	-0.053123	-0.281313	0.515177	0.144295	0.861261	-0.547586	0.449939	...	0.858940	0.171893	0.213212	0.724704	0.282572	-0.184230	0.047207	0.777123	0.200676	0.197031
2	0.469994	1.000000	0.192589	0.478979	-0.069871	0.376007	0.360065	-0.009070	-0.048613	-0.106673	...	0.776622	-0.661464	0.261766	0.737797	0.645086	-0.263735	0.783299	-0.077354	0.759020	0.723939
3	-0.550472	0.192589	1.000000	0.193281	-0.356091	-0.761879	0.424255	-0.796791	0.949131	-0.924456	...	-0.428467	-0.839177	-0.406396	-0.235516	0.071931	-0.629252	0.694929	-0.521313	0.682164	0.419160
4	-0.053123	0.478979	0.193281	1.000000	-0.164855	0.184677	-0.539969	-0.408731	0.076561	-0.350036	...	0.346865	-0.548084	0.814321	0.645531	-0.265524	0.192633	0.190750	-0.144999	0.215360	0.847099
5	-0.281313	-0.069871	-0.356091	-0.164855	1.000000	0.586600	-0.062786	-0.099317	-0.531213	0.625730	...	-0.056786	0.378236	0.067706	-0.333228	0.410334	0.758564	-0.173813	-0.596997	-0.464576	-0.506735
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
606	-0.184230	-0.263735	-0.629252	0.192633	0.758564	0.669408	-0.686380	0.033711	-0.694202	0.652328	...	0.022494	0.563759	0.584298	-0.054997	-0.176084	1.000000	-0.624226	-0.236262	-0.790867	-0.324706
607	0.047207	0.783299	0.694929	0.190750	-0.173813	-0.189345	0.720539	-0.376038	0.493797	-0.515529	...	0.259973	-0.867517	-0.286430	0.239912	0.675320	-0.624226	1.000000	-0.369820	0.949711	0.560176
608	0.777123	-0.077354	-0.521313	-0.144999	-0.596997	0.102755	-0.155415	0.829896	-0.322770	0.249925	...	0.428138	0.382644	0.152201	0.460790	-0.316263	-0.236262	-0.369820	1.000000	-0.099893	0.035601
609	0.200676	0.759020	0.682164	0.215360	-0.464576	-0.297534	0.674080	-0.241977	0.552247	-0.606777	...	0.316475	-0.873060	-0.258381	0.367724	0.509017	-0.790867	0.949711	-0.099893	1.000000	0.663637
610	0.197031	0.723939	0.419160	0.847099	-0.506735	-0.026222	-0.098944	-0.290294	0.309695	-0.559918	...	0.486695	-0.803084	0.507949	0.759342	-0.027031	-0.324706	0.560176	0.035601	0.663637	1.000000

Figure 5.5: User Similarity per item Genre

6 Rating Prediction for Test User:

Model has been evaluated for test user (user_*id*=2000) which shows movie$_{id}$, title and corresponding prediction for each movie predicted by model according to genre of the movie as shown in Fig 5.6

	user_id	movie_id	rating	prediction	title	genres
0	2000	1639	5	0.003029	Chasing Amy (1997)	Drama|Romance
1	2000	2529	5	0.000633	Planet of the Apes (1968)	Action|Sci-Fi
2	2000	1136	5	-0.004764	Monty Python and the Holy Grail (1974)	Comedy
3	2000	2321	5	-0.003958	Pleasantville (1998)	Comedy
4	2000	2858	5	-0.001914	American Beauty (1999)	Comedy|Drama
5	2000	2501	5	-0.005929	October Sky (1999)	Drama
6	2000	2804	5	-0.005575	Christmas Story, A (1983)	Comedy|Drama
7	2000	1688	5	-0.005625	Anastasia (1997)	Animation|Children's|Musical
8	2000	1653	5	-0.001556	Gattaca (1997)	Drama|Sci-Fi|Thriller
9	2000	527	5	-0.008114	Schindler's List (1993)	Drama|War
10	2000	1619	5	-0.009162	Seven Years in Tibet (1997)	Drama|War
11	2000	110	5	-0.003253	Braveheart (1995)	Action|Drama|War
12	2000	1193	5	0.007835	One Flew Over the Cuckoo's Nest (1975)	Drama
13	2000	318	5	0.005677	Shawshank Redemption, The (1994)	Drama

Figure 5.6: Prediction Rating for Test User (*user_id*=2000)

7 Recommend Movies:

Finally top recommended movies has shown for test user(user_id=2000) which slows movie_id, title and corresponding prediction for each movie predicted by model according to genre of the movie as shown in Fig 5.7.

	movie_id	prediction	title	genres
0	1450	0.036005	Prisoner of the Mountains (Kavkazsky Plennik) ...	War
1	2557	0.034918	I Stand Alone (Seul contre tous) (1998)	Drama
2	2283	0.026346	Sheltering Sky, The (1990)	Drama
3	832	0.024870	Ransom (1996)	Drama|Thriller
4	1444	0.024722	Guantanamera (1994)	Comedy
5	2269	0.023707	Indecent Proposal (1993)	Drama
6	3025	0.023672	Rough Night in Jericho (1967)	Western
7	3744	0.023366	Shaft (2000)	Action|Crime
8	1729	0.023357	Jackie Brown (1997)	Crime|Drama
9	2578	0.023318	Sticky Fingers of Time, The (1997)	Sci-Fi
10	1947	0.023302	West Side Story (1961)	Musical|Romance
11	3275	0.023082	Boondock Saints, The (1999)	Action|Comedy
12	3626	0.022973	8 1/2 Women (1999)	Comedy
13	1825	0.022870	Player's Club, The (1998)	Action|Drama

Figure 5.7: Recommended Movies for Test User (*user_id*=2000)

8 Trust Computation Result:

By observing all above results, the MSE (Mean Square Error) values are decreasing for DNN, DNN with Trust, GDCF and GDCF with Trust model respectively. When trust has been incorporated with DNN the MSE value has improved (see Fig. 5.8).

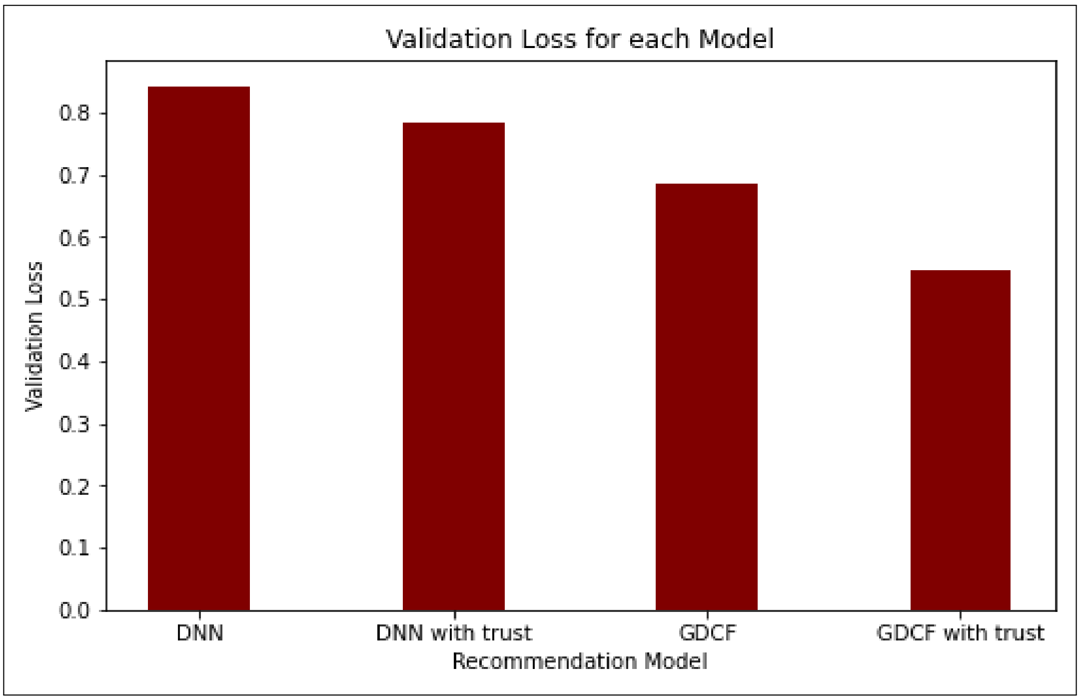

Figure 5.8: Performance Comparison of Proposed Model and Existing Model

Chapter 6

Conclusion

This work is aimed to focus on improving the recommendation accuracy of the previous DNN model to address data sparsity problem. The recommendation model is being developed using metadata of item (genre) with deep neural network techniques and a PCC based trust computation method which combines user similarity with weighted trust propagation. A trust computational model has been developed that permits only trusted similar profiles whose similarity exceeds with some predefined threshold value.

And to solve the cold start problem, we propose a new personalized recommendation model known as Genre Based Deep Collaborative Filtering (GDCF) which takes inputs as user id, movie id and item genre as metadata for a better understanding of the latent features representation, which is usually not captured by the user item rating matrix. The performance of proposed model is estimated using Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE). The experimentation of GDCF with trust model is carried out on MovieLens dataset which shows a major improvement over the baseline methods with high accuracy of 81% with 0.60 MSE value.

REFERENCES

[1] Xue, F., He, X., Wang, X., Xu, J., Liu, K., & Hong, R. (2019).Deep item-based collaborative filtering for top-n recommendation. ACM Transactions on Information Systems (TOIS), 37(3), 1-25

[2] Pan, Y., He, F., & Yu, H. (2019). A novel enhanced collaborative autoencoder with knowledge distillation for top-N recommender systems. Neurocomputing, 332, 137-148.

[3] Zhang, L., Luo, T., Zhang, F., & Wu, Y. (2018). A recommendation model based on deep neural network. IEEE Access, 6, 9454-9463.

[4] Zhou, W., Li, J., Zhang, M., Wang, Y., & Shah, F. (2018). Deep learning modelling for top-n recommendation with interests exploring. IEEE Access, 6, 51440-51455.

[5] uang, T., Zhao, R., Bi, L., Zhang, D., & Lu, C. (2021). Neural embedding singular value decomposition for collaborative filtering. IEEE Transactions on Neural Networks and Learning Systems.

[6] Shoja, B. M., & Tabrizi, N. (2019). Customer reviews analysis with deep neural networks for e-commerce recommender systems. IEEE access, 7, 119121-119130.

[7] Y. L. J. L. a. W. L. Bo Yang,"Social Collaborative Filtering by Trust," Pattern Analysis and Machine Intelligence, Nov 2016.

[8] Wu, Z., Tian, L., Zhang, Y., & Wang, Z. (2021). Web User Trust Evaluation: A Novel Approach Using Fuzzy Petri Net and Behavior Analysis. Symmetry, 13(8), 1487.

[9] Choudhury, S. S., Mohanty, S. N., & Jagadev, A. K. (2021). Multimodal trust based recommender system with machine learning approaches for movie recommendation. *International Journalof Information Technology, 13(2), 475-482.*

[10] Wu, X., Yuan, X., Duan, C., & Wu, J. (2019). A novel collaborative filtering algorithm of machine machine learning by integrating restricted Boltzmann

machine and trust information. Neural Computing and Applications, 31(9), 4685-4692.

[11] Li, Y., Liu, J., Ren, J., & Chang, Y. (2020). A novel implicit trust recommendation approach for rating prediction. IEEE Access, 8, 98305-98315.

[12] Deng, S., Huang, L., Xu, G., Wu, X., & Wu, Z. (2016). On deep learning for trust-aware recommendations in social networks. IEEE transactions on neural networks and learning systems, 28(5), 1164-1177.

[13] Guo, L., Liang, J., Zhu, Y., Luo, Y., Sun, L., & Zheng, X. (2019). Collaborative filtering recommendation based on trust and emotion. Journal of Intelligent Information Systems, 53(1), 113-135.

[14] Ma, X., Lu, H., Gan, Z., & Zeng, J. (2017). An explicit trust and distrust clustering based collaborative filtering recommendation approach. Electronic Commerce Research and Applications, 25, 29-39.

[15] Papagelis, M., Plexousakis, D., & Kutsuras, T. (2005, May). Alleviating the sparsity problem of collaborative filtering using trust inferences. In International conference on trust management (pp. 224-239). Springer, Berlin, Heidelberg.

[16] Xiao, Y., Pei, Q., Liu, X., & Yu, S. (2018). A novel trust evaluation mechanism for collaborative filtering recommender systems. IEEE access, 6, 70298-70312.

[17] Choudhury, S. S., Mohanty, S. N., & Jagadev, A. K. (2021). Multimodal trust based recommender system with machine learning approaches for movie recommendation. International Journal of Information Technology, 13(2), 475-482.

[18] O'Donovan, J., & Smyth, B. (2005, July). Trust no one: Evaluating trust-based filtering for recommenders. In IJCAI (Vol. 5, pp. 1663-1665).

[19] He, X., Liao, L., Zhang, H., Nie, L., Hu, X., & Chua, T. S. (2017, April). Neural collaborative filtering. In Proceedings of the 26th international conference on world wide web (pp. 173-182).

[20] Lu, J., Wu, D., Mao, M., Wang, W., & Zhang, G. (2015). Recommender system application developments: a survey. Decision Support Systems, 74, 12-32.

[21] Wang, J., Zhu, L., Dai, T., & Wang, Y. (2020). Deep memory network with Bi-LSTM for personalized context-aware citation recommendation. Neurocomputing, 410, 103-113. doi:10.1016/j.neucom.2020.05.04

LIST OF PUBLICATIONS ON PRESENT WORK

[1] Sandeep A. Thorat, Ashwini Gavade, Seema Mane and Sourabh Bakshi "Research Challenges, Opportunities and Applications in Collaborative Filtering and Content-based Recommendation System", National Conference on Advances in Science, Engineering and Technology for Sustainable Development (NCASETSD - 2022), (Status: Best Paper Award and extended to IEEE)

[2] Ashwini Hakke, Ajit Mali, "A Trust Based Collaborative Filtering Recommendation System using Deep Neural Network", IEEE Transactions on Learning Technologies. (Status: Under Review)

Keywords: Recommendation System, Content-based, Collaborative Filtering, Research Challenges, Applications

Sandeep A. THORAT[0000-0002-1641-0376]*,

Ashwini GAVADE[0000-0003-4104-845X]*,

Seema MANE[0000-0001-8377-974X]*,

Sourabh BAKSHI[0000-0001-7245-9230]*

Research Challenges, Opportunities, and Applications in Collaborative Filtering and Content-based Recommendation Systems

Abstract

Due to the exponential growth of data on Internet, providing correct information to users in a reasonable amount of time has become a challenge. The recommender system acts as an information filtering tool; they provide appropriate information as per the user's choice and interest. In recent years, it has become a widespread technique that is being used in many applications. In general, recommendation systems have been classified into collaborative and content-based filtering. Due to its significance, the recommender system has become one of the most important research areas in today's context. Though recommendations systems are being used for a quite long time, many research challenges and issues in the design of effective recommendation systems are yet to be addressed in an effective manner. The purpose of this paper is to familiarize people with the research challenges and opportunities in recommender systems and their available solutions. The collaborative filtering approach stills suffer from many drawbacks, including data sparsity, gray sheep, cold start problem, and scalability. The content-based filtering approach suffers from reciprocity, sparsity and limited content analysis issues. This paper reviews existing research approaches to overcome these challenging issues. We have also discussed future research directions in collaborative filtering and content-based recommendation systems. Various application domains have listed out where recommendation systems can be improved, such as healthcare, agriculture, etc. The paper describes possible future extensions in all these applications. Overall this paper will act as a guide for those who are interested in doing research in the recommendation system.

* Computer Science Department, Rajarambpau Institute of Technology, Sangli, India,

sandip.thorat@ritindia.edu, ashwinihakke@gmail.com, seema.chavan1012@gmail.com, bakshisourabh@gmail.com

Government College of Engineering, Karad

(An Autonomous Institute of Government of Maharashtra)

National Conference on

Advances in Science, Engineering and Technology for Sustainable Development (NCASETSD - 2022)

20th - 21st May, 2022

Sponsored by IEEE

CERTIFICATE

This is to certify that, Prof./Dr./Mr./Ms. S.Thorat, A.Gavade, S.Mane, S.Bakshi *has participated and presented a paper titled* Research Challenges, Opportunities & Applications in Collaborative Filtering and Content based Recommedation Systems *in National Conference on* ***"Advances in Science, Engineering and Technology for Sustainable Development (NCASETSD-2022)"*** *on* ***20th - 21st May, 2022.***

Dr. S. R. Kurode
Convenor

Dr. A. T. Pise
Program Chair

A Trust Based Collaborative Filtering Recommendation System using Deep Neural Network

Journal:	*Transactions on Learning Technologies*
Manuscript ID	Draft
Manuscript Type:	Regular paper
Keywords:	Trust, Collaborative Filtering, Deep Learning

A Trust Based Collaborative Filtering Recommendation System using Deep Neural Network

Ashwini Hakke, *student Member, IEEE*, Ajit Mali, *student Member, IEEE*

***Abstract*—In the Era of Big Data, users have to deal with large volume of data due to e-commerce websites, social media etc. Due to this exponential growth of data, fetching the right information from this information pool is a difficult task. A recommendation system (RS) deals with this problem and acts like an information filtering tool and filters the most important information based on data provided by the user as per his/her preference and interest.Collaborative Filtering Recommendation System (CF) is one of the widely used RS approach which is based on collecting and exploring information about users' past behavior in terms of their likes and dislikes. Even though CF is one of the successful and commonly used RS still it suffers from different problems such as cold-start, data sparsity, scalability and Gray Sheep Problem. To address data sparsity problem, we proposed Genre Similarity Based Trust Model which considers genre specific user similarity and permits only trusted similar profiles whose similarity exceeds with some predefined threshold value for each genre. And to solve the cold start problem, we proposed a new personalized recommendation model known as Genre Based Deep Collaborative Filtering (GDCF) which takes inputs as user item rating matrix and item genre as metadata for a better understanding of the latent features representation, which is usually not captured by the user item rating matrix. The performance evaluation of proposed model is estimated using Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE) on MovieLens dataset which shows a major improvement over the baseline methods with high accuracy of 81% with 0.60 MSE value.**

***Index Terms*—Collaborative Filtering, Deep Neural Network, Recommendation System, Trust**

I.INTRODUCTION:

Many different domains have used recommender systems, including product suggestions, articles, book recommendations, movie recommendations, music recommendations, news recommendations, and web page and document recommendations. Recommender Systems (RSs) gather data on user preferences for a selection of goods (e.g., movies, jokes, applications, websites, travel destinations etc) [19]. A user's behaviour, such as the songs they listen to, the apps they download, the websites they visit, and the books they read, can be tracked explicitly (usually by collecting ratings given by users) or implicitly (often by tracking their online activity). RS may make use of user demographic information (like age, nationality, gender) In Web 2.0, social data like followers, followed, twits, and postings are frequently employed. Use of data from the Internet of things is becoming more prevalent (e.g., GPS locations, RFID etc.)[10]. Recommender systems are built using different methods. Depending on the information utilized to make recommendations, these systems are divided into Content based (CB) filtering or Collaborative Filtering (CF). CB techniques are based on profile of the target user's past interests the description of items. A CF technique is generally based on the notion that persons with similar preferences about certain items are likely to have alike preferences for other items [5].

Although Collaborative Filtering RS is the most popular recommendation system, it still has some flaws like data sparsity (If only small portion of items are rated by active users then it's difficult to find sufficient number of similar users), cold start problem (It represents the difficulty of making recommendations for new users and items), Scalability(If the volume of dataset is increased, then system is unable to generates satisfactory recommendations), and Graysheep problem (Gray sheep occurs in CF systems where the preferences and taste of a user do not equate with any group and ultimately, the recommendation process fails to yield good results)[8]. In actuality, comparing the ratings that two users provide for different items is a way to determine how comparable they are. The number of products that each user must rate in order for their ratings to be comparable is typically low, hence a minimum number of corated items must have been rated by both users.

Due to these limitations of CF recommendation system, trust and similarity between users are taken into account to raise the level of suggestion rather than only similarity-based users. Now a days, people prefer to acquire their information from people they can trust, like parents, friends,

Author Dashboard

- 1 Submitted Manuscripts
- Start New Submission
- Legacy Instructions
- 5 Most Recent E-mails

Submitted Manuscripts

STATUS	ID	TITLE	CREATED	SUBMITTED
ADM: Arnold, Joyce • Under review	TLT-2022-09-0231	A Trust Based Collaborative Filtering Recommendation System using Deep Neural Network View Submission	07-Sep-2022	07-Sep-2022

Contact Journal

K.E. Society's

Rajarambapu Institute of technology, Rajaramnagar

An Autonomous Institute

(Affiliated to Shivaji University)

SYNOPSIS OF M.TECH DISSERTATION

1.	**Name of Program**	:	M.Tech (Computer Science & Engineering)
2.	**Name of Student**	:	Ashwini Chandrakant Hakke (2030008)
3.	**Date of Registration**	:	June 2021
4.	**Name of Guide**	:	Dr. S. A. Thorat
5.	**Sponsored details (if any)**		
6.	**Proposed Title**	:	**"A Trust Based Collaborative Recommendation System using Deep Neural Network"**
7.	**Synopsis of dissertation work:**		

7.1 Relevance

A recommendation system is a subclass of Information filtering Systems that seeks to predict the rating or the preference a user might give to an item. The recommender system deals with a large volume of information present by filtering the most important information based on the data provided by a user and other factors that take care of the user's preference and interest. It finds out the match between user and item and imputes the similarities between users and items for recommendation. In simple words, it is an algorithm that suggests relevant items to users. There are majorly three types of recommender systems which work primarily in the Media and Entertainment industry: Collaborative Recommender system, Content-based recommender system and Knowledge based recommender system.

Collaborative Recommender System:

It's the most sought after, most widely implemented and most mature technologies that is available in the market. Collaborative recommender systems aggregate ratings or recommendations of objects, recognize commonalities between the users on the basis of their ratings, and generate new recommendations based on inter-user comparisons. The greatest strength of collaborative techniques is that they are completely independent of any machine-readable representation of the objects being recommended and work well for complex objects where variations in taste are responsible for much of the variation in preferences. Collaborative filtering is based on the assumption that people who agreed in the past will agree in the future and that they will like similar kind of objects as they liked in the past.

Content based Recommender System:

It's mainly classified as an outgrowth and continuation of information filtering research. In this system, the objects are mainly defined by their associated features. A content-based recommender learns a profile of the new user's interests based on the features present, in objects the user has rated. It's basically a keyword specific recommender system here keywords are used to describe the items. Thus, in a content-based recommender system the algorithms used are such that it recommends users similar items that the user has liked in the past or is examining currently.

Knowledge based Recommender System:

This type of recommender system attempts to suggest objects based on inferences about a user's needs and preferences. Knowledge based recommendation works on functional knowledge: they have knowledge about how a particular item meets a particular user need, and can therefore reason about the relationship between a need and a possible recommendation.

7.2 Techniques

1. Memory based approach:

Memory-Based Collaborative Filtering approaches can be divided into two main sections: user-item filtering and item-item filtering. A user-item filtering takes a particular user, find users that are similar to that user based on similarity of ratings, and recommend items that those similar users liked. In contrast, item-item filtering will take an item, find users who liked that item, and find other items that those users or similar users also liked. It takes items and outputs other items.

2. Model-based Approach:

Model-based Collaborative filtering approach involves building a model based on the dataset of ratings. In other words, we extract some information from the dataset, and use that as a "model" to make recommendations without having to use the complete dataset every time. This approach potentially offers the benefits of both speed and scalability.

Matrix Factorization:

Matrix Factorization is a technique to discover the latent factors from the ratings matrix and to map the items and the users against those factors.

Singular Value Decomposition:

The Singular Value Decomposition (SVD), a method from linear algebra that has been generally used as a dimensionality reduction technique in machine learning. SVD is a matrix factorization technique, which reduces the number of features of a dataset by reducing the space dimension from N-dimension to K-dimension (where K¡N). In the context of the recommender system, the SVD is used as a collaborative filtering technique. It uses a matrix structure where each row represents a user, and each column represents an item. The elements of this matrix are the ratings that are given to items by users.

Back propagation with DNN:

Back-propagation is just a way of propagating the total loss back into the neural network to know how much of the loss every node is responsible for, and subsequently updating the weights in such a way that minimizes the loss by giving the nodes with higher error rates lower weights and vice versa.

Neural networks are machine learning algorithms used to model complex patterns in datasets. The layers of interconnected functional units called nodes or neurons are nonlinear activation functions. In this model the input layer of the network is an embedding layer (a fully connected layer represents the sparseness to a dense vector) where the user and item feature vector are input to it.

Deep Neural Network:

DNN is the multiple neural building blocks that can be composed in to a single differentiable and trained the network end to end. The interesting features are end to end differentiability and provide suitable inductive bias catered to the input data. Data complexity and large training samples leads to reasonable performance gain. The deep learning method tried to predict the rating of an item for an active user and recommend an item to a user accordingly. The embedding layer receives the user-item feature matrix as an input. The dot product of two feature matrices are passed through dense hidden layer in which, number of nodes in the layer is equal to the number of features.

Trust based filter:

In CF, Consumer searches for the information and producer provides the information. The technical hitches of CF systems can be dealt with trust properties. The trust is assessed as readiness to accept the truth in a client on the basis of expertise and performance in a particular time. A trust system is considered as a network of interfacing peers. The communication between two domestic entities is considered as an end product of building the trust association with an additional entity by the prescribed representations of trust.

7.3 Present Theories and Practices

In [1], author proposed a novel trust measure matrix which combines user similarity with weighted trust propagation to address cold start problem, data sparsity and malicious attacks. Non cold user passed through different models with trust filter and a cold user generated an optimal score with their preferences for recommendation. The proposed model recommends the movie to the trusted user on the basis of implicit trust value and different machine learning approaches.

The Matrix Factorization is the most common model based technique to find the embeddings or features that makes up the interest of a particular user. In this paper author also give various machine learning techniques such as Singular Value Decomposition (SVD), Back propagation, Deep Neural Network (Auto encoder) model were implemented to predict the accurate recommendation.

Matrix factorization is a process, where the actual users' rating matrix is divided into two feature (user and item) matrices in such a way that multiplication of the new two matrices will give the actual matrix. The predicted matrix has similar output with the true values, and the 0 ratings in actual matrix are replaced with the prediction based on the similar users' preferences. SVD proposes prediction of the rating of an item for a target user that is not yet rated and if the predicted rating is high then that item will be recommended to the target user. To generate a recommendation for a target user feature vector of the target user from the newly computed user-item rating matrix has been considered and select high predicted rating items.

In Back propagation model the input layer of the network is an embedding layer (a fully connected layer represents the sparseness to a dense vector) where the user

and item feature vector are input to it. User Feature embedding will supply the weights for user, which will be placed between input layer and hidden layer. Item Feature embedding will supply the weights for item, which will be placed between hidden layer and output layer. The network weights were updated according to the error back propagated with number of epochs while trained and test the network. The network updated itself in a way of stochastic gradient descent method. The dot product of randomly generated user-item feature vector provides the predictive rating as an output.
The deep neural network method tried to predict the rating of an item for an active user and recommend an item to a user accordingly. The embedding layer receives the user-item feature matrix as an input. The dot product of two feature matrices are passed through dense hidden layer in which, number of nodes in the layer is equal to the number of features. Accuracy of the model is measured through MSE (Mean square error).
A trust based filter system is considered as a network of interfacing peers. The communication between two domestic entities is considered as an end product of building the trust association with an additional entity by the prescribed representations of trust. According to Donovan and Smyth, Local trust and Global trust are two important distinction of trust metric. Without peer dependency, computation of a single score termed as Global Trust. Local trust metric provides personalized scores, means it suggests trustable peer.

This [2] paper proposes a novel social recommendation method combined with a restricted Boltzmann machine model and trust information to improve the performance of recommendations which alleviated data sparsity and cold start problem. Specifically, users' preferences and ratings of items are used as data inputs in a restricted Boltzmann machine model to learn the probability distribution. Author first propose a model based collaborative filtering recommender system using restricted Boltzmann machine (RBM), and then incorporate social network information in RBM-based CF systems by adjusting the predicted ratings of certain items according to the ratings of trusted users. To get better results and increase the speed of learning, the momentum method is used when updating the parameters.

In addition, user similarities are calculated by weighting user similarity and user trust values derived from trust information (i.e., trust statements explicitly given by users). A new method is proposed to calculate the trust value given the direct binary trust value between two users, if two users with great similarity trust each other, they tend to have similar preferences. The main idea of the method is to find the users trusted by a specific user and modify the predicted ratings given by the RBM model using the ratings of trusted users. Predictions are made by integrating user-history ratings and ratings of trusted users from a well-trained restricted Boltzmann machine model.

In [3] Author gives a new implicit trust recommendation approach (ITRA) to generate item rating prediction by mining and utilizing user implicit information in recommender systems. Specifically, user trust neighbor set that has similar preference and taste with a target user is first obtained by trust expansion strategy via user trust diffusion features in a trust network. Then, the trust ratings mined from user trust neighbors are used to compute trust similarity among users based on user collaborative filtering model. Finally, using the filtered trust ratings and user trust similarity, the prediction results are generated by a trust weighting method.
This model uses the trust similarity of user entity acquired by mining implicit trust in the systems for recommendations. However, the systems may still have some trusted users with low similarity. Hence, to improve rating prediction accuracy, a trust weighting model is designed by considering the inferred trust value and user trust similarity.

In this paper [4], Author developed a novel trust-based approach for recommendation in social networks. The major benefit of this approach is the ability of dealing with the problems with cold-start users. In addition to social networks, user trust information also plays an important role to obtain reliable recommendations. Although matrix factorization (MF) becomes dominant in recommender systems, the recommendation largely relies on the initialization of the user and item latent feature vectors.
In particular, Author attempt to leverage deep learning to determinate the ini-

tialization in MF for trust-aware social recommendations and to differentiate the community effect in user's trusted friendships. A two-phase recommendation process is proposed to utilize deep learning in initialization and to synthesize the users' interests and their trusted friends' interests together with the impact of community effect for recommendations. He utilized a deep learning method, i.e., deep autoencoder, which is an efficient approach to nonlinear dimensionality reduction. Deep autoencoder is quite suitable for the initialization phase, which can abstract the high-dimensional rating data to latent features through multiple layers of restricted Boltzmann machines (RBMs.)

In this paper [5] Author presented a collaborative filtering recommendation method based on trust and emotion , which can improve the recommendation accuracy in the case of data sparsity, effectively resist shilling attacks, and achieve high recommendation accuracy for cold start users. First method is based on explicit and implicit satisfaction to alleviate the sparsity problems. Second, establish trust relationships among users using objective and subjective trust. Objective trust is determined by similarity of opinion, including rating similarity and preference similarity. Third, based on the trust relationship, a set of trusted neighbors is obtained for a target user. Next, to further exclude malicious users or attackers from the neighbors, the set is screened according to emotional consistency among users, which is mined from implicit user behavior information. Finally, based on the ratings of items by the screened trusted neighbors and the trust relationships among the target user and these neighbors, to obtain a list of recommendations for the target user.
To overcome the sparsity problem of the user-item rating matrix, implicit satisfaction is introduced. If explicit satisfaction does not exist between user and an item, the implicit satisfaction of user of rating based on the similarity. Based on the explicit satisfaction degree and the implicit satisfaction degree, the user-item satisfaction degree matrix is calculated. In social networks, if two users have similar hobbies and personality traits or are close and familiar with each other, the trust level is typically high. Thus, the items that are recommended by these two users are more acceptable to other users. Therefore, enhanced trust is divided into objective trust, which is generated by similarity of opinion, and subjective trust,

which is generated by familiarity.

In this paper [6] Author proposed a novel method that works to improve the performance of collaborative filtering recommendations by integrating sparse rating data given by users and sparse social trust network among these same users to address data sparsity and cold start problem. This is a model-based method that adopts matrix factorization technique that maps users into low-dimensional latent feature spaces in terms of their trust relationship, and aims to more accurately reflect the user's reciprocal influence on the formation of their own opinions and to learn better preferential patterns of users for high-quality recommendations. A novel matrix factorization based social CF method named TrustMF which is Simple but effective way to map users into two low-dimensional spaces, truster space and trustee space, by factorizing trust networks according to the directional property of trust. The vectors of truster (B) and trustee (W) in two spaces describe "to trust by browsing" and "to be trusted by writing" behaviors of a user, respectively. The key idea behind the proposed truster and trustee models is to construct a bridge between ratings and trust by mapping them into the same d-dimensional latent space.

In this paper [7] An explicit trust and distrust clustering based collaborative filtering recommendation method is proposed. Firstly, a SVD signs based clustering algorithm is proposed to process the trust and distrust relationship matrix in order to discover the trust communities. Secondly, a sparse rating complement algorithm is proposed to generate dense user rating profiles which alleviates the sparsity and cold start problems to a very large extent. Finally, the prediction of missing ratings can be obtained by combining the newly generated user rating profiles and the traditional collaborative filtering algorithm. A SVD signs based trust community mining method to partition users into different communities according to how they trust and distrust others. SVD signs clustering method is a kind of Spectral Graph partitioning method which is derived from the Fiedler method.

In this paper [8] An author proposed a scheme called RHT (recommendation from high trust value entities) to evaluate the trust degree of service recommended by high trust value entities. RHT first selects the trusted ones from those users by computing the similarity between target user and them. Simultaneously, RHT also withstands malicious attacks during the trusted nodes selection. In addition, an author has designed an adaptive trust computation method to calculate trust value according the ratings of trusted users.

First, compute the similarity between nodes from three aspects, i.e., common-friend, common-interest and common-BC. Finally, we combine these similarity on these three aspects by assigning different weight coefficients to derive a final similarity. Resistance mechanism to prevent a malicious node from disguising as a trusted node. Finally, compute direct and indirect trust value by combining the similarity with ratings, respectively.

7.4 Proposed Work

The work is proposed to study existing methods used for developing trust based collaborative recommendation systems and perform comparative analysis of these methods. As part of this work we will select an effective method from existing system and develop a recommendation system. This work will attempt to design a novel trust based collaborative recommendation system using deep neural network and compare the performance of same with existing systems.

Objectives

1. To perform literature survey on trust based collaborative recommendation system and identify gap for new work.
2. To study and implement an existing trust based collaborative recommendation system using machine learning.
3. Proposing novelty in existing machine learning trust based collaborative recommendation system using deep neural network to achieve improvement.
4. To perform the performance comparison of existing machine learning trust based collaborative recommendation system and the proposed trust based collaborative recommendation system.

Possible Outcomes:

1. Identification and study of various methods for recommendation system

2. Implementation of novel trust based collaborative recommendation system using deep neural network which give better experience to the user.

Proposed work is planned in following phases.

Phase I- Literature Survey and Synopsis Preparation: Duration: Nov 2021 – Dec 2021

In this phase we aim to do the Literature survey by collecting different journal papers. On that basis:

- Identify and study various methods of trust based collaborative recommendation system.
- Identifying various challenges of current trust based collaborative recommendation system.
- Using literature survey, preparing the synopsis for the dissertation work.

Phase II- Implement an existing trust based collaborative filtering using machine learning, Report writing and submission. Duration: Jan 2022 – Feb 2022

In this phase we plan to carry out following task.

- To find existing machine learning techniques to design a trust based collaborative system.
- To perform critical analysis and identify merits and demerits of the system and work on possible refinements in these system.

Phase III- Implementation of the proposed system and collecting experimental observations.

Duration: March 2022 – May 2022

In this phase we plan to carry out following task.

- To implement proposed trust based collaborative recommendation system using deep neural network.
- Collect experimental observations.

Phase IV- Comparison and analysis of experimental results of proposed method and earlier method. Report writing and submission.

Duration: June 2022 –July 2022

In this phase we plan to carry out following task.

- Comparison and analysis of proposed trust based collaborative recommendation system and existing trust based collaborative recommendation system.

8 Facilities Required for Project:

To carry out dissertation work, facilities that will be needed are mentioned below:

- Operating System: Windows
- Programming Language: Python

9 Expected Date for Completion of Work: June 2022

10 Approximate Expenditure: Nil

Date: Ashwini Hakke

Place: RIT, Rajaramnagar Student

Dr. S. A. Thorat	Dr. S. S. Patil	Dr. N. V. Dharwadkar
Guide,	HOP,	HOD,
Dept. of CSE	Dept. of CSE	Dept. of CSE
R.I.T. Rajaramnagar	R.I.T. Rajaramnagar	R.I.T. Rajaramnagar

Bibliography

[1] S. N. M. K. J. Sasmita Choudhury1, “Multimodal trust based recommender system with machine,” International Journal Of Information Technolgy, pp. 475-482, April 2021.

[2] X. Y. C. D. J. W. Xiaojun Wu, “A novel collaborative filtering algorithm of machine learning by integrating restricted Boltzmann machine and trust information,” Neural Computing and Applications, April 2018.

[3] J. L. J. R. ,. A. Y. C. YAKUN LI, “A Novel Implicit Trust Recommendation Approach for Rating Prediction,” IEEE Access, vol. 8, 2020.

[4] L. H. G. X. X. W. a. Z. W. Shuiguang Deng, “On Deep Learning for Trust-Aware Recommendations in Social Networks,” IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, pp. 50-62, 2016.

[5] J. L. Z. Y. L. L. S. X. Z. Liangmin Guo, “Collaborative filtering recommendation based on trust and emotion,” J Intell Inf Syst, july 2018.

[6] Y. L. J. L. a. W. L. Bo Yang, “Social Collaborative Filtering by Trust,” Pattern Analysis and Machine Intelligence, Nov 2016.

[7] H. L. Z. G. J. Z. Xiao Ma, “An Explicit Trust and Distrust Clustering based Collaborative Filtering Recommendation Approach,” Electronic Commerce Research and Applications, June 2017.

[8] Q. P. X. L. A. S. Y. YANG XIAO1, “A Novel Trust Evaluation Mechanism for Collaborative Filtering Recommender,” IEEEAccess, Feb 2017.

PAPER NAME

plagarism_2030008_Report.docx

AUTHOR

Ashwini Hakke

WORD COUNT

11811 Words

CHARACTER COUNT

66451 Characters

PAGE COUNT

61 Pages

FILE SIZE

1.5MB

SUBMISSION DATE

Nov 23, 2022 12:51 PM GMT+5:30

REPORT DATE

Nov 23, 2022 12:52 PM GMT+5:30

10% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 10% Publications database
- Crossref database
- Crossref Posted Content database

Excluded from Similarity Report

- Internet database
- Submitted Works database
- Bibliographic material
- Quoted material
- Cited material
- Small Matches (Less then 20 words)

10% Overall Similarity

Top sources found in the following databases:

- 10% Publications database
- Crossref database
- Crossref Posted Content database

TOP SOURCES

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Donghui Wang, Yanchun Liang, Dong Xu, Xiaoyue Feng, Renchu Guan. "... Crossref	3%
2	"Soft Computing for Problem Solving", Springer Science and Business ... Crossref	1%
3	Braja Gopal Patra, Vahed Maroufy, Babak Soltanalizadeh, Nan Deng, W.... Crossref	<1%
4	Sandeep A. Thorat. "Payload Content based Network Anomaly Detectio... Crossref	<1%
5	Jieun Son, Seoung Bum Kim. "Content-based filtering for recommenda... Crossref	<1%
6	Kai Wang, Tiantian Zhang, Tianqiao Xue, Yu Lu, Sang-Gyun Na. "E-com... Crossref	<1%
7	Alaa Alslaity, Thomas Tran. "Evaluating Recommender Systems", Inter... Crossref	<1%
8	Benard M. Maake, Sunday O. Ojo, Tranos Zuva. "chapter 5 Information ... Crossref	<1%
9	Tham Vo. "An integrated network embedding with reinforcement learni... Crossref	<1%

10	P. Venkateswara Rao, A. P. Siva Kumar. "The societal communication ... Crossref	<1%
11	Akshi Kumar, Simran Seth, Shivam Gupta, Shubham. "Sentiment-Enha... Crossref	<1%
12	Jyoti Shokeen, Chhavi Rana. "An Application-oriented Review of Deep ... Crossref	<1%
13	"Intelligent Techniques for Web Personalization", Springer Science and... Crossref	<1%
14	Matjaž Kragelj, Mirjana Kljajić Borštnar. "Automatic classification of ol... Crossref	<1%
15	Muhammad Asim, Shah Khusro. "Content Based Call for Papers Reco... Crossref	<1%
16	"Design, User Experience, and Usability. Design for Contemporary Inter... Crossref	<1%
17	"Big Data and Networks Technologies", Springer Science and Business ... Crossref	<1%
18	"Trends in Artificial Intelligence Theory and Applications. Artificial Intel... Crossref	<1%
19	Kristian Wahyudi, Johanes Latupapua, Ritchie Chandra, Abba Suganda ... Crossref	<1%